

Development of Segmentation-Rendering on Virtual Reality for Training Deep-learning, Simulating Landscapes and Advanced User Experience

Tomohiro Fukuda¹, Marcos Novak², Hiroyuki Fujii³

¹Osaka University ²University of California Santa Barbara ³Forum8 Co., Ltd.

¹fukuda@see.eng.osaka-u.ac.jp ²marcos@mat.ucsb.edu ³fujii@forum8.co.jp

Virtual reality (VR) has been suggested for various purposes in the field of architecture, engineering, and construction (AEC). This research explores new roles for VR toward the super-smart society in the near future. In particular, we propose to develop post-processing rendering, segmentation-rendering and shadow-casting rendering algorithms for novel VR expressions to enable more versatile approaches than the normal photorealistic red, green, and blue (RGB) expressions. We succeeded in applying a wide variety of VR renderings in urban-design projects after implementation. The developed system can create images in real time to train deep-learning algorithms, can also be applied to landscape analysis and contribute to advanced user experience.

Keywords: Super-smart society, Virtual Reality, Segmentation, Deep-learning, Landscape simulation, Shader

INTRODUCTION

The so-called “super-smart society” is approaching in the near future through such innovations as industry 4.0 and society 5.0 [1, 2]. In this society, a huge amount of information (big data) from sensors and the Internet of Things devices in physical space will be accumulated in virtual space. Artificial intelligence will analyze this data, and the results will be fed back to humans in physical space in various forms. Virtual reality (VR), which has features of presence, interaction, and autonomy (Zeltzer, 1992), and was selected one of the 14 Grand Challenges for Engineering in the 21st Century [3], has been suggested for various purposes such as design study, presentation, simulation and communication in the field

of architecture, engineering, and construction (AEC) (Dokonal et al. 2016, Dorta et al. 2016, Natephra et al. 2017, Eloy et al. 2018, Fukuda et al. 2019).

This research explores new roles for VR in the super-smart society. In particular, we propose to develop post-processing rendering, segmentation-rendering, and shadow-casting rendering algorithms for novel VR expressions to enable more versatile approaches for training deep-learning, simulating landscapes and advanced user experience than the photorealistic red, green, and blue (RGB) expressions that are normally used in VR.

SITUATIONS AND PROBLEMS

Toward the realization of super-smart society, the situations and problems examined in this research are as follows:

Self-driving cars and smart-inspection, which are projected to be outputs of the super-smart society, are inseparable from AEC fields. In order to realize these technologies, various kinds of training must be performed on self-driving/smart-inspection systems (Kuwata et al. 2009, Yabuki et al. 2018). Training using physical space and physical objects is limited in terms of the various types that can be prepared, and costs tend to become high. By contrast, training using virtual space and virtual objects readily increases the types of training data that can be used and tends to reduce cost.

Deep-learning has become a popular subset of machine learning due to its high level of performance and applied to the AEC field (Cao et al. 2019). A great way to use deep-learning to segment objects in images is to build a convolutional neural network (CNN). This CNN needs a large amount of training data (images in this study) for learning. In fact, photographs are often used, but it is not easy to collect large amounts of photographs. For learning, in addition to a normal RGB image, its paired foreground/background (binary) and parts-based (categorical) shape images of it are also required. Using a 3D virtual model in VR makes it easy to change parameters such as camera information and objects' properties, thereby making it possible to readily and automatically create a large amount of image data for learning if the 3D model's accuracy is high. On the other hand, a system for simply generating foreground/background (binary) and parts-based (categorical) shape images is needed (Problem-A).

Via the concept of a "digital twin," which is closely related to the super-smart society, digital copies of physical buildings and urban spaces can be used for real-time optimization (Söderberg et al. 2017). VR is then expected to act as an interface that presents information to users in physical space via an easy-to-understand method with various expressions during

feedback. When simulating landscapes in architectural and urban-design process, design targets are normally studied in the VR space expressed in photorealistic RGB. On the other hand, it is easier to understand by visualizing according to analytical targets such as the distance from the main viewpoint to the design target (visual distance: short-distance view, middle-distance view, long-distant view) and the shadow-falling duration per point (Problem-B).

PROPOSED VR SYSTEM

In order to tackle the problems described in Chapter 2, we develop post-processing rendering, segmentation-rendering, and shadow-casting rendering algorithms. Regarding segmentation-rendering method, VR images can be divided into regions and categories corresponding to different objects or parts of objects in real-time.

We developed new rendering functions on UC-win/Road, which is a VR software for which rendering customization function is provided (Figure 1 (a)). Generally, it can execute rendering processing on the screen multiple times, which is usually only once. It can also implement pre-processing and post-processing of the whole rendering, and pre-processing and post-processing of one scene rendering respectively. Furthermore, it can define variable values to be delivered to the shader when rendering objects, which can be used on the shader processing side. These functions can be implemented as plug-in software and can be switched on VR (Figure 2).

In order to realize a segmentation algorithm for each type of 3D object constituting a VR application, the types of 3D objects are divided into *Road*, *River*, *Intersection*, *Model*, *RoadSideModel*, *Character*, *Car*, *Sign*, and so on. In these types, *Road* is further subdivided into *Guardrail*, *Road*, *Bridge*, *Tunnel*, *Sidewalk*, etc., as a lower hierarchy. Similarly, *Model* is further comprised of *Vehicle*, *Building*, *Structure*, *TrafficLight*, *Plant*, etc. as a lower hierarchy (Figure 1 (b)).

Figure 1
(a) Rendering customization feature: The general processing flow, (b) Object definition for Segmentation

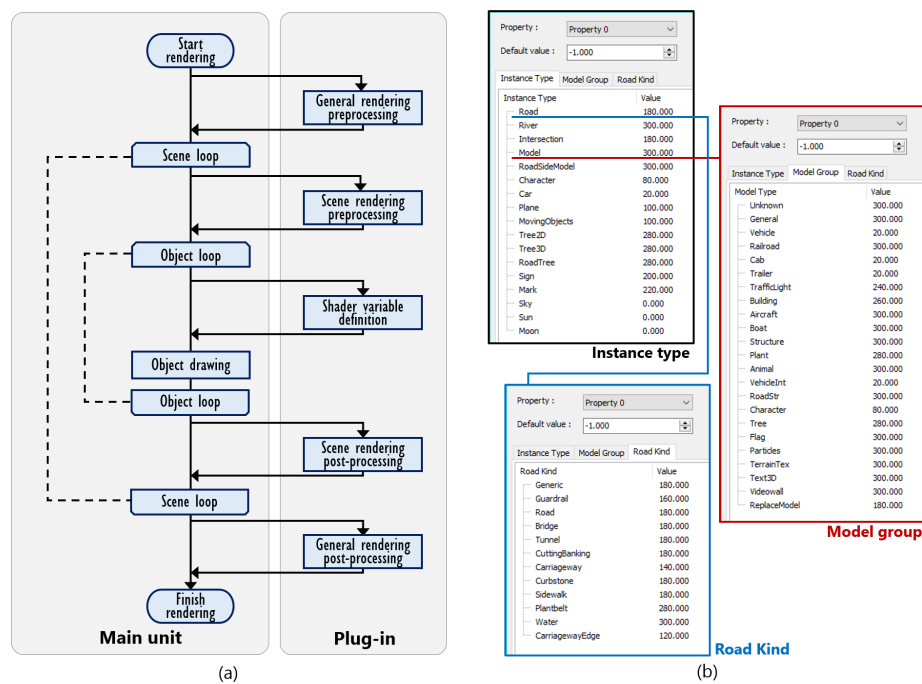
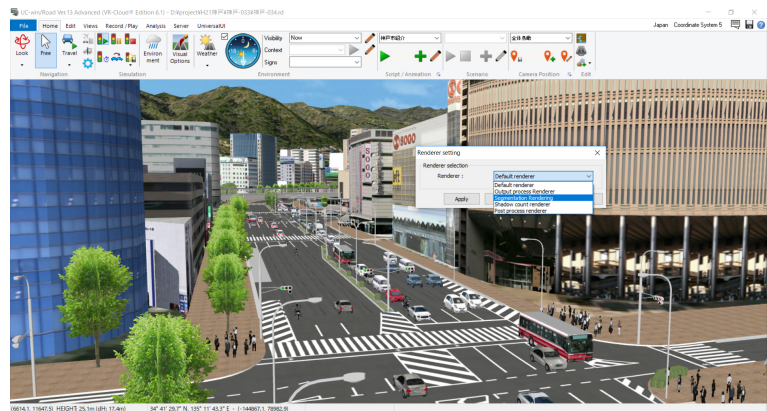
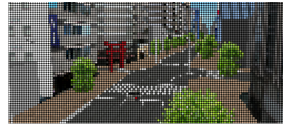
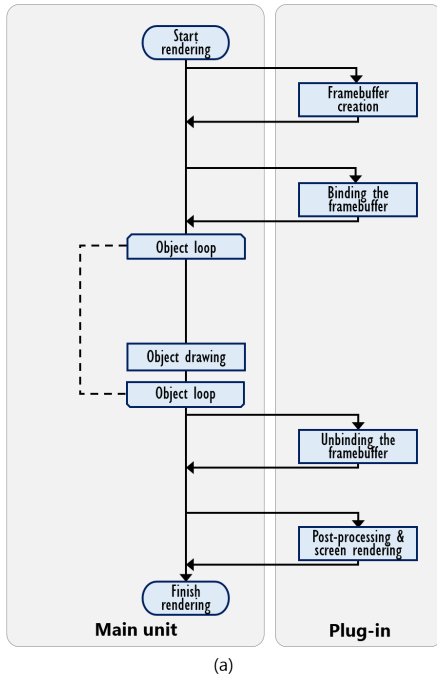


Figure 2
Switching rendering options on VR





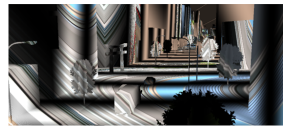
(b)



(c)



(d)



(e)

Figure 3
Post-processing: (a)
Whole processing
flow, (b) Ball, (c)
Focus, (d)
Monochrome, (e)
Artistic

Post-processing rendering

First, we implemented post processing as the simplest rendering customization processing (Figure 3). In the pre-processing, rendering is performed in a framebuffer which is a temporary rendering destination, and post-processing is performed from the rendering result and output to the screen. The shader program is coded by GLSL (OpenGL Shading Language) and can be customized by the user.

Segmentation rendering

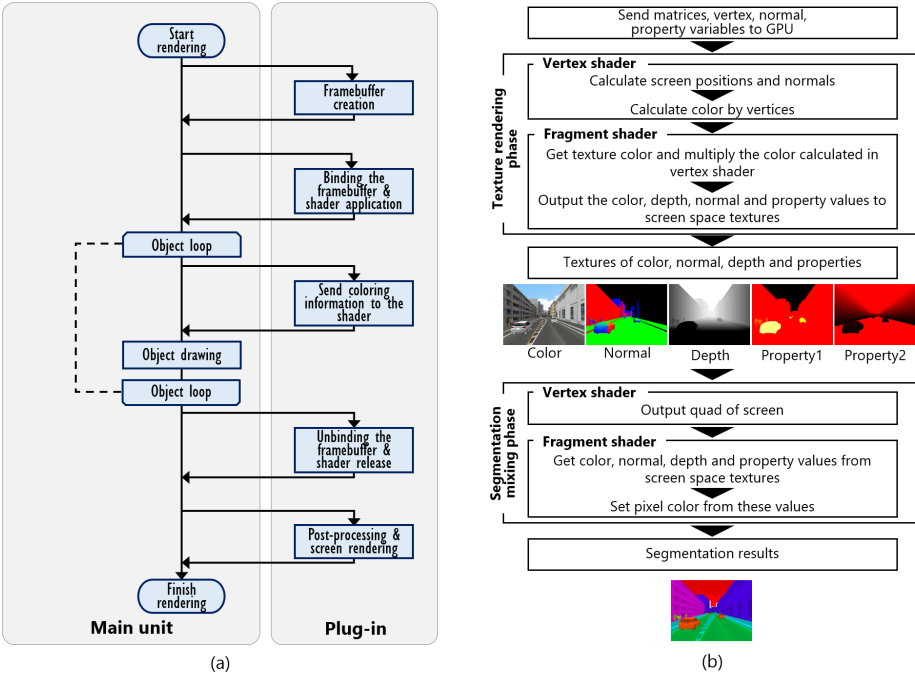
We implemented a plug-in that segments objects and colors each object to generate materials for various machine learning and for landscape analysis (Figure 4 (a)).

In pre-processing of the entire rendering, create a framebuffer to be a temporary rendering destination, and apply a shader programmed to write color

information, normal information, depth information, and property values of the scene to five textures. Next, in the object drawing process, the hue value according to the type of the objects, speed information, acceleration information, distance from the viewpoint, height information from the terrain is calculated and delivered as the property values on the plug-in. We implemented the processing to calculate the final output using the rendered texture in post-processing of the entire rendering. According to this implementation, it is possible to switch and execute segmented coloring, speed, acceleration, distance from the viewpoint, and height from the ground from ordinary RGB representation by changing the setting of post-processing.

The internal-rendering flow of the proposed segmentation rendering is described as follows (Figure 4 (b)). The input data (matrices, vertices, normals, and

Figure 4
Segmentation
rendering: (a)
Whole processing
flow, (b) Rendering
flow



property values) are sent to the GPU (Graphics Processing Unit). In the first *texture-rendering phase*, the *vertex-shader process* consists of calculating screen positions and normals and then calculating color by vertices. The *fragment-shader process* consists of obtaining texture color, multiplying the color calculated in the vertex shader, and then outputting the color, depth, normal, and property values to screen-space textures. In the next *segmentation mixing phase*, the *vertex-shader process* consists of outputting quad of the screen. The *fragment-shader process* consists of obtaining colors, normals, depths, and property values from screen-space textures and then setting pixel colors based on these values.

Shadow-casting rendering

In the VR city model, we implemented multi-shadowing as a plug-in software to simulate the ef-

fects of shadow-falling a day (Figure 5 (a)).

In normal shadow rendering processing, the scene color and the shadow color are mixed and displayed. On the other hand, in the proposed system, the scene color is not used, and the part where the shadow falls is black and the part where the sun shines is white (Figure 5 (b)). Next, time-varying scenes are rendered multiple times and their rendered results are blended. Then, it was rendered that the part with many suns is white and the part with many shadows is black, and it was temporarily written in the framebuffer (Figure 5 (c)). Finally, the calculation results are colored and output to the screen as a heat map in order to clearly show the effects of sunshine in the post-processing (Figure 5 (d)).

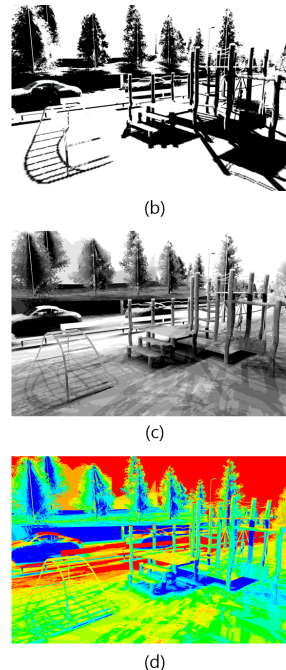
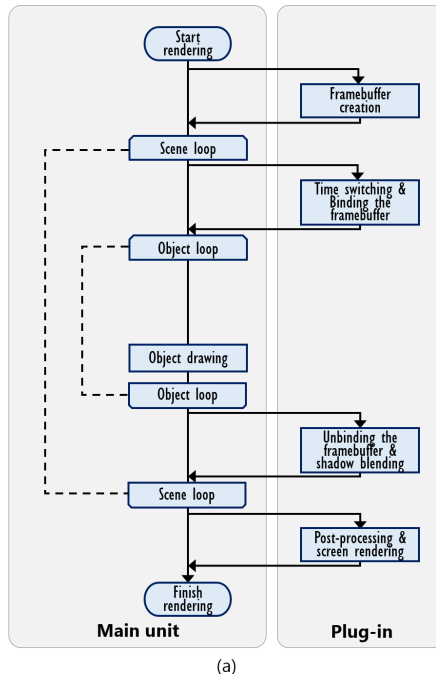


Figure 5
Shadowing: (a)
Whole processing
flow, (b) One
shadow rendering,
(c) Blended image
with multiple
shadow rendering
results, (d) Heat
map by
post-processing

RESULTS AND DISCUSSION

We applied the developed system to VR urban-design applications and realized the following wide variety of renderings (Figure 6). Various VR expressions with real-time rendering and user interaction were made possible based on the large-scale 3D-VR model.

Post-processing rendering is output by shaders as a set of ball-like expressions, focused expressions, monochrome expressions, posterization expressions, artistic expressions, etc., which will contribute to the advanced user experience.

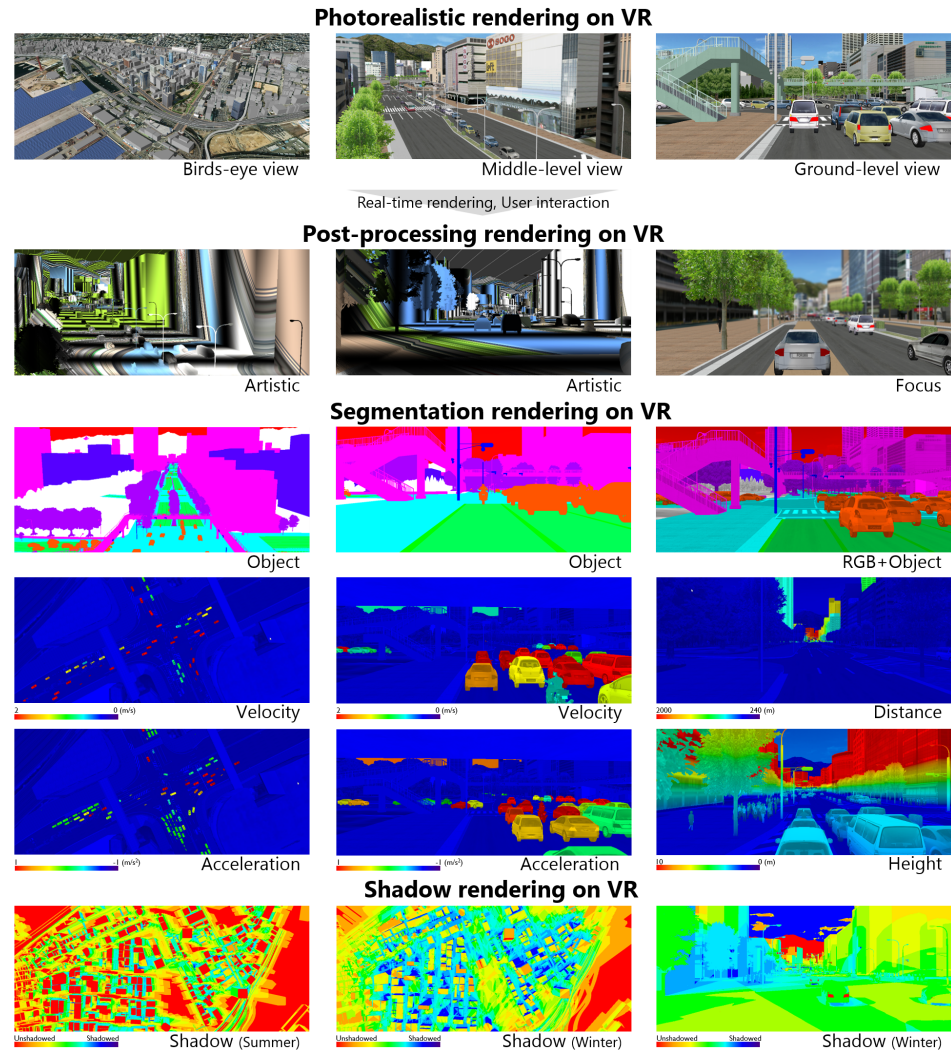
Object Segmentation involves segmentation by each type of 3D object described in Chapter 3 (for Problems A and B). Using this VR rendering method, foreground/background (binary) and parts-based (categorical) shape images can be easily created by changing the camera viewpoint (automation

is possible). *RGB + Object Segmentation* is expressed by mixing normal photorealistic RGB and segmentation representations based on the types of 3D objects described in Chapter 3 (for Problems A and B). A user can experience VR while simultaneously looking at the appearance of the object and its type.

Velocity Segmentation involves segmentation according to the speeds of moving 3D objects such as cars and pedestrians (for Problems A and B). *Acceleration Segmentation* involves segmentation according to the acceleration of moving 3D objects such as cars and pedestrians (for Problems A and B). In normal RGB expression, it is difficult to understand the velocity and acceleration of moving objects such as cars and pedestrians. On the other hand, this VR expression makes it possible to intuitively understand the velocity and acceleration of moving objects.

Distance Segmentation involves segmentation

Figure 6
Developed wide
variety of
renderings on VR



according to the distance from the VR camera, which is the viewpoint of pedestrians and drivers who operate VR (for Problems A and B). *Height Segmentation* involves segmentation according to the height from

the ground level of the terrain (for Problem B). When considering landscape design, it is important to understand the distance from the viewpoints of each element that consists of the urban landscape. The ele-

ments that compose the landscape look different as short-distance view, middle-distance view, and long-distance view depending on the distance from the viewpoint. The intuitive examination has been difficult in the conventional photorealistic rendering. On the other hand, this VR expression makes it possible to intuitively understand the distance from the viewpoint to each element and the height from the ground surface.

Shadow-count rendering involves expression according to shadowing duration to each point in the VR space at summer and winter solstices, respectively (for Problem B). This VR expression is a tool that can perform more sophisticated shadow simulation and can support more detailed design.

CONCLUSIONS

This research contributes to the development of post-processing rendering, segmentation-rendering and shadow-casting rendering algorithms for novel VR expressions by presenting situations and problems related to the super-smart society. We succeeded in applying a wide variety of rendering outputs to VR applications of urban-design projects. The developed system can easily create foreground/background (binary) and parts-based (categorical) shape images in addition to a normal photorealistic RGB image to train deep-learning algorithms, can also be applied to landscape simulation according to analysis targets such as the distance from the main viewpoint to the design targets and the shadow-falling duration per point in VR urban space, and contribute to advanced user experience.

ACKNOWLEDGMENTS

This research was done with one of the research activities from 2017 to 2018 by World16 (Yoshihiro Kobayashi, Kostas Terzidis, Marc Aurel Schnabel, Paolo Fiamma, Amar Bennadji, Thomas Tucker, Dongsoo Choi, Matthew Swarts, Ruth Ron, Taro Nara-hara, Wael Abdelhameed, Marcos Novak, and Tomohiro Fukuda), an international research group on virtual reality. The authors wish to acknowledge the

support received from Forum8 Co. Ltd.

REFERENCES

- Cao, R, Fukuda, T and Yabuki, N 2019 'Quantifying Visual Environment by Semantic Segmentation Using Deep Learning', *Proceedings of CAADRIA 2019*, Wellington, pp. 71-80, Vol.2
- Dokonal, W, Knight, M and Dengg, E 2016 'VR or Not VR - No Longer a Question?', *Proceedings of eCAADe 2016*, Oulu, pp. 573-579, Vol.2
- Dorta, T, Kinayoglu, G and Hoffmann, M 2016, 'Hyve-3D and the 3D Cursor: Architectural co-design with freedom in Virtual Reality', *International Journal of Architectural Computing*, 14(2), pp. 87-102
- Eloy, S, Ourique, L, Woessner, U, Kieferle, J and Schotte, W 2018 'How Present am I - Three virtual reality facilities testing the fear of falling', *Proceedings of eCAADe 2018*, Lodz, pp. 717-726, Vol.2
- Fukuda, T, Yokoi, K, Yabuki, N and Motamedi, A 2019, 'An Indoor Thermal Environment Design System for Renovation Using Augmented Reality', *Journal of Computational Design and Engineering*, 6(2), p. 179-188
- Kuwata, Y, Teo, J, Fiore, G, Karaman, S, Frazzoli, E and How, JP 2009, 'Real-Time Motion Planning with Applications to Autonomous Urban Driving', *IEEE Transactions on Control Systems Technology*, 17(5), pp. 1105-1118
- Natephra, W, Motamedi, A, Fukuda, T and Yabuki, N 2017, 'Integrating building information modeling and virtual reality development engines for building indoor lighting design', *Visualization in Engineering*, pp. 5-19
- Söderberg, R, Wärmefjord, K, Carlson, JS and Lindkvist, L 2017, 'Toward a Digital Twin for real-time geometry assurance in individualized production', *CIRP Annals*, 66(1), pp. 137-140
- Yabuki, N, Nishimura, N and Fukuda, T 2018 'Automatic Object Detection from Digital Images by Deep Learning with Transfer Learning', *Advanced Computing Strategies for Engineering. EG-ICE 2018 (25th Workshop of the European Group for Intelligent Computing in Engineering). Lecture Notes in Computer Science*, Lausanne, pp. 3-15, Vol.10863
- Zeltzer, D 1992 'Autonomy, Interaction, and Presence', *Presence*, pp. 127-132, Vol.1
- [1] <http://alvarestech.com/temp/tcn/CyberPhysicalSystems-Industrial4-0.pdf>
- [2] https://www8.cao.go.jp/cstp/english/society5_0/index.html
- [3] <http://www.engineeringchallenges.org/challenges.aspx>