Visualizing 3D form Using SketchTo3D Tool

Balgum Song[†], Chul Soo Kim[††]

ABSTRACT

Numerous studies have attempted to present converting 2D drawings to 3D. However, converting 3D shapes to exactly how a person thinks is challenging because 2D drawings include numerous variables and possibilities. This study focuses on visualizing 2D to 3D in a commonly used 3D animation software required in animation education or the 3D industry. We implemented our SketchTo3D tool to add the editing and automatic texture assigning method from the imported 2D image into the 3D software that previously had to be done manually. As a result, the SketchTo3D tool saves time to immediately visualize the composition, shape, and volume to express the 3D character, providing an opportunity to break down the barrier between 2D and 3D.

Key words: 2D, 3D Conversion, Modeling, Texturing, Visualizing

1. INTRODUCTION

1.1 Background and Purpose

Drawing a picture can quickly create an intuitive form with a small number of lines. However, drawings must consider the perspective, angle, and focal length, which isn't simple to express three-dimensional visual shapes. Therefore, computer graphics development using 3D specialized software is useful for expressing three-dimensional shapes. However, learning 3D software to express a three-dimensional specific shape requires professional knowledge acquisition and time and effort [1]. Nevertheless, the usage of 3D specialized software is increasing owing to the possibility of visual creation and a wide range of requirements.

Furthermore, many experts have been steadily introducing research to provide and convert 3D information from 2D sketches. However, 2D expression is complex to implement in 3D because it contains countless variables and possibilities to

shape it visually in 3D [2]. Therefore, work is carried out separately between 2D designers and 3D producers, which is inefficient in many animation and VFX industries.

Architecture and Engineering tend to have 3D converter software to create a 3D object from 2D easily. Still, the design and animation industry does not have such an efficient automated 3D converter which requires manual work. In architecture, many types of software such as AutoCAD, ArchiCAD, CEDREO, Civil, Draft it, Floorplanner, Foyr Neo, Google sketch-up, Homebyme, Planner 5D, Revit, Roomle, Vectorworks, etc. easily convert into 3D. However, architecture Software requires skills and precise operation to convert into 3D. In addition, architectural software tends to specialize in building a floor plan for construction, and many restrictions to creating animated characters, such as people, animals, and monsters.

The VFX and film industries also require research that can be directly implemented and viewed

※ Corresponding Author: Chul Soo Kim, Address: (48305) 302-301, 100, Gwanganhaebyeon-ro, Suyeong-gu, Busan, Korea, TEL: +82-51-629-5361, E-mail: kimcsoo@pknu.ac.kr
Receipt date: Aug. 8, 2022, Revision date: Oct. 3, 2022
Approval date: Oct. 25, 2022

[†] International College Animation & VFX., Dongseo University (E-mail: sbu1977@dongseo.ac.kr)
[††] Dept. of Industrial Design., Pukyong National University

in 3D visualization. Therefore, in this study, we developed a SketchTo3D conversion, focusing on design and animation industries that differ from architecture. The SketchTo3D tool will visualize 3D vision to expand the design quality. In addition, the SketchTo3D tool is expected to use widely in the 3D industry and educational institutions without any special equipment or additional software.

1.2 Scope and Methods

Our approach in this study is to enable automatic processes from 2D to 3D, focusing on the Maya software, which is popular in the VFX and animation industries. We used the latest version of Maya 2023 software and a python program to enhance the tool functions.

The operation settings are simplified, even for unskilled users to visualize the 3D output immediately. First, load the saved image file into the executed SkethTo3D tool and then require three steps according to the characteristics of the image. Drawing at least a two-sided view will automatically create the final 3D work. These simple processes will automatically create a textured 3D modeling character used in the 3D industries. The entire process is shown in Fig. 1.

Chapter 2 examines the latest tools studied for the 2D to 3D conversion. Chapter 3 discusses the process and how the SketchTo3D conversion tool works. In Chapter 4, we implement input drawings

to the SketchTo3D and see the result of the generated modeling and texturing output. Finally, Chapter 5 explores the effects and problems and analyzes the solutions.

2. RELATED STUDIES

Recently, various studies have been commonly developed in visualizing sketch design or 2D photo images into 3D. Among them, the latest research techniques for 3D visualization are introduced and compared with this study.

First, Monster Mash [3] only requires a simple viewpoint to create a character in a 3D view. This novel inflation process method, similar to Teddy [4], does not require a complex and tedious approach with accurate measurement. The other benefit of Monster Mash is setting an animation movement without a rigging system for the character. Unlike many other studies containing rigging system approaches with controllers such as RigMesh [5], Arti Sketch [6], and AniMesh [7], this simplified process is ideal for non-experts to create movements. However, Monster Mash has drawbacks. Creating a 3D modeling with appropriate depth on each proportion can be challenging from a single view and require additional modification to the image closer to the boundary areas. Due to the inflating method, Monster Mash is used for simplified round characters drawn directly from sketches and is unsuitable for constructions and

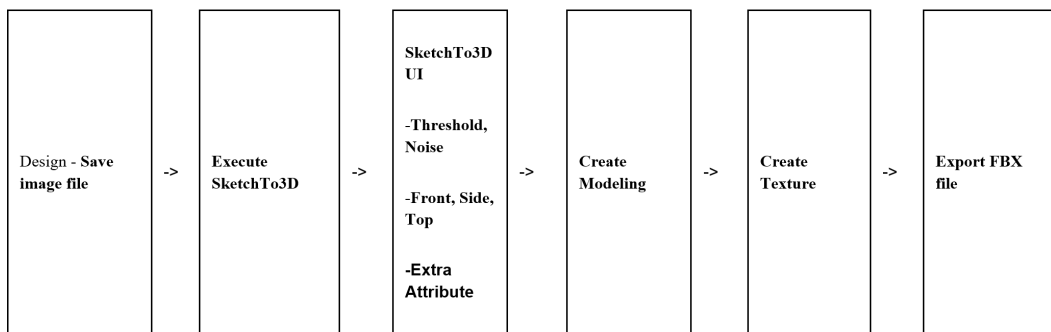| Design - Save image file	->	Execute SketchTo3D	->	SketchTo3D UI -Threshold, Noise -Front, Side, Top -Extra Attribute	->	Create Modeling	->	Create Texture	->	Export FBX file

Fig. 1. The SketchTo3D tool process.

other complex rigid objects.

Second, PIFuHD [8] uses the pixel-aligned implicit function to create the 3D character shape. A single image predicts the backside by the MLP prediction network, a valuable tool to create a 3D clothed human easily. The downside is limited to a single human character without a complex pose, which is unsuitable for sketch and design.

Third, Air scaffolds [9] contain a pen drawing with a hand motion to visualize in 3D perspective view. Combining those two enables precise drawings consisting of 3D shapes rather than drawing alone without a hand motion. This novel approach is useful to work simultaneously with 2D with the 3D virtual space but requires equipment and training to familiarize with this study.

Third, unlike the previous accurate input sketches, Wang et al. [10] challenge working with sketch data and have proposed a unique approach to converting 3D point cloud construction based on 2D freehand sketches. They applied the PSGN [11] method with a standardization module to process sketches in various drawing styles. However, with a noisy point cloud system is impossible to create accurate details and lead to an unexpected 3D shape.

Recently, deep neural networks have become the primary process of creating image-based modeling. Usually, the system works by creating depth maps to create the depth of the character. Then, the point cloud system is reconstructed based on a dran character, resulting in the 3D mesh. These state-of-the-art techniques perform better 3D results and quality but are limited to a specific depth and characters. Our study is based on how to work efficiently with simple sketches and drawings with fewer limitations to visually see it in 3D view in the 3D animation and film pipeline.

3. METHOD

3.1 Adding Image Editing System in Maya

2D editing software such as Adobe Illustrator or Photoshop has a wide range of functions for choosing the shape of a drawn character. However, it is difficult to identify each character on the plane where the outer boundary is to extract an image character with a limited amount of function in 3D software. 3D software is optimized for CG production, but difficult to edit images and drawings.

AOV Render in Maya has an 'Object ID' option to separate each object assigned as a unique color for compositing purposes. Still, this option is not suitable for extracting image characters. Therefore, we approached a new method by adding an image editing function to the 3D software. Color AtPoint′s[12] built-in command in Maya can pick out the color information on the image UV 0 to 1 space, which can be useful to extract a character by some certain color value. However, when dividing countless points on an image in the 0 to 1 space, a wide ratio interval lowers to identify the character′s outline border. In contrast, a narrow ratio interval increases the accuracy of detecting the border edge but will cause a slow operation speed to perform various tasks. Therefore, instead of implementing Maya's built-in function, we used the OpenCV findContours method [13], which saves time and provides precise results. We added three specific arguments in the findContours method. The first argument is the binary image we transfer into black and white. Then applied, RETR_EXTERNAL for the second argument, which works mainly with getting the outer border of an image. Finally, include the CHAIN_APPROX_SIMPLE argument to get the list of image (x,y) coordinates of boundary angular points for the third argument, which works best with our method. Then we connect the findContours method to the Maya command intSliderGroup, which toggles the threshold and noise to achieve the best contour results.

SketchTo3D UI configuration only requires three settings to create 3D characters. First, achieve the accurate border result by toggling the Threshold and Noise sliders. The imported image is automati-

cally converted into a curve displaying the border in Maya. However, depending on the character shape and image color values, it may not correctly detect the border. With a heuristic approach, we set the default setting of Threshold and Noise to 185 and 12, respectively, but might require additional settings if needed. Threshold captures a black and white color range from 0 to 255 levels (Fig. 2.), and noise removes the crooked outlines. There is a limit to the computer's ability to automatically recognize the front, top, and side of a character. Therefore, users need to manually select the appropriate outline curves for each side and click the Front, Side, and Top buttons to input them. This tool will create the 3D output character with at least choosing two directions. Users can also set the character's texture resolution for the final look. Finally, execute the Convert 2D to 3D button to process the 3D object.

3.2 Assigning Textures to the Output Modeling

Generating the input image in 3D is divided into two procedures: modeling implementation and texture application. First we extrude the boundary curves for each direction, and if each direction's shape does not match, we place a bounding box around the relevant forms and adjust the average size to match each direction. Then, we implement Maya's Boolean intersection technique to get the output modeling. Next, Maya's Automatic mapping function is applied to implement color information. Unfortunately, in Maya software, there is no simple way to wrap textures to a 3D object from a single orthographic image drawing. Therefore, we tried various ways to apply the multiview textures from a single image. First, we tested projecting each direction by implementing Maya's Layered Texture and Blend color nodes to bring out the texture output. Still, the texture of the outline part rotates to the side and stretches when a texture is projecting from the front, side, and top Fig 3. The result did not come out well because the overlapping parts of the front, side, and top surfaces have limited options to fix the stretched colors. As a solution, we added a function by slightly increasing the size in each direction to cover the stretched texture area and came out with a better result. Fig. 3 red, green and blue faces show the thickness preventing from showing those stretched textures.

4. RESULT

4.1 Hard Surface Character

We tested six hard surface sketch image references (Table 1): Ship [14], Yacht [15], JetSki, Boat, Battle Cruiser [16], and SkySanctuary [17]. JetSki and Boat design originated from a student(Zhang Hanwen), and the rest images are from the internet to achieve the final 3D process. The final 3D shapes implementing boolean functions are hard to deal with smooth forms and cause sharp edges instead of round, smooth shapes[18]. Therefore, these hard surface characters came out with better shapes than the organic characters because the hard surface is more machine-made sharp objects that are
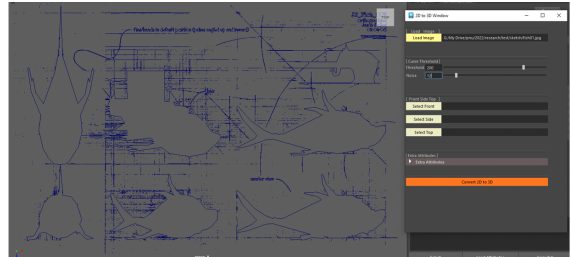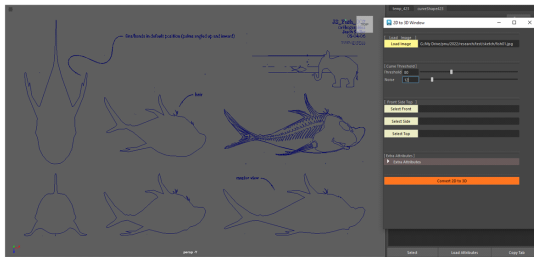

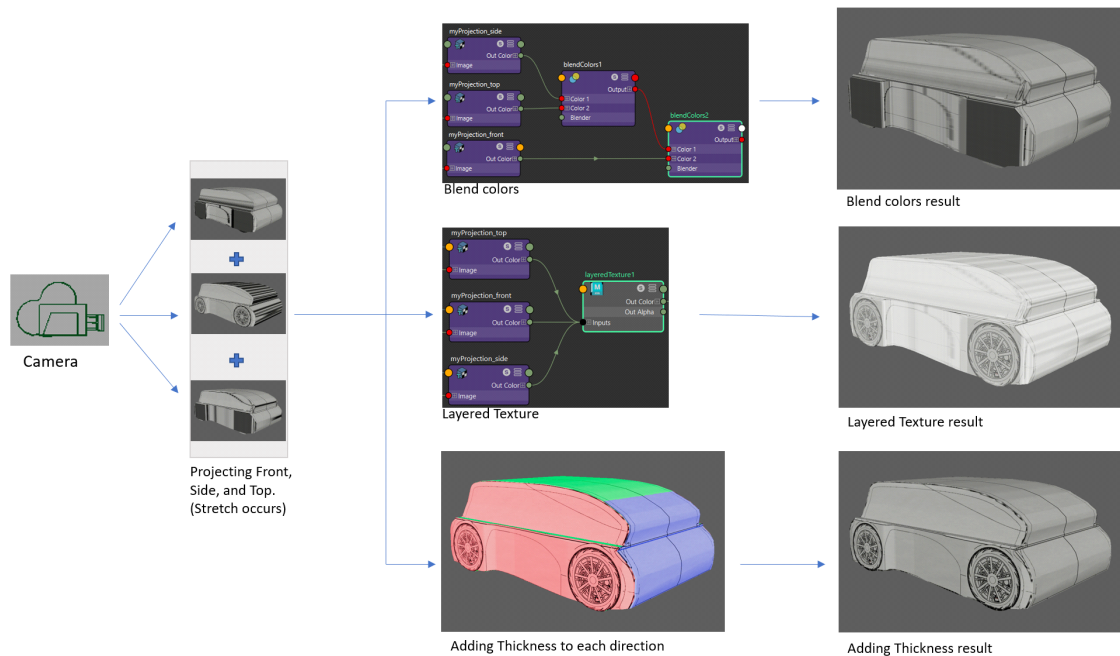Fig. 2. Set Threshold to 80 left and 200 right.

Fig. 3. Texture Projection.

better with boolean operations. However, small details are not expressed accurately because of the limitation to the blind spot areas with only three side views.
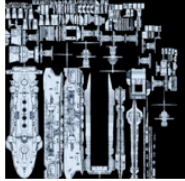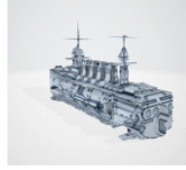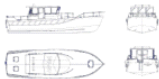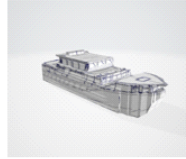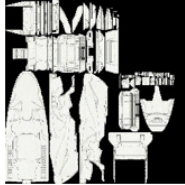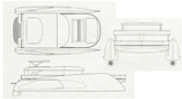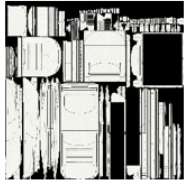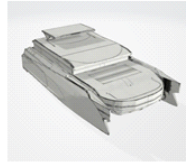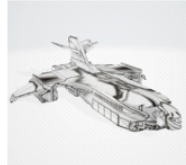
4.2 Organic Character

We tested a total of six organic characters, all referenced from the internet (Table 2.): Divoer01 [19], Diver02 [20], Goat [21], Turtle [22], Fish01 [23], and Fish02 [24]. With the SketchTo3D tool, we only applied the front and the side view to the Diver01 and Diver02 characters because they do not have a top view in their drawings. Since it is created with only the front and side, the shape's range is minimal, and some proportions' thickness tends not to come out properly. In addition, the Goat and Fish01 characters tend to have additional eyes created due to the lack of information in overlapping elements by projection mapping. Organic characters usually have smooth and round surfaces but come out in a rough shape due to Maya's boolean limited function.

5. DISCUSSION AND CONCLUSION

3D software such as Maya, 3ds Max, and Blender provides 3D production output but does not automatically create 3D results based on a drawing. Therefore, many industries produce 3D production manually from the 2D source, which takes a sufficient amount of time and cost. Furthermore, many works should be modified or recreated because the final 3D production result may look different from the original concept of the design. Many designers in the 3D industries tend to be creative with fewer limitations[25], but many previous studies are limited to converting 3D shapes from specific photos or characters. Unlike previous studies, this study is expected to contribute to the production pipeline by providing the SketchTo3D tool to visualize the 3D output from the 2D source automatically. SketchTo3D tool, with its editing function, extracts the border edge from a character and converts it to 3D output, then add textures from the 2D source image, making the process easier and more efficient with minor operations.
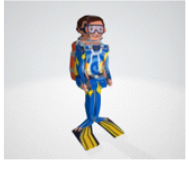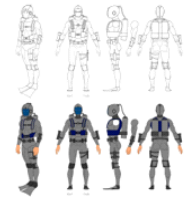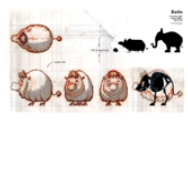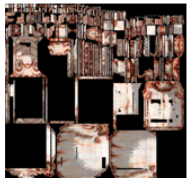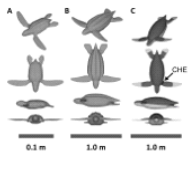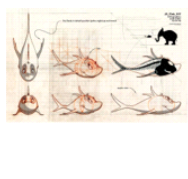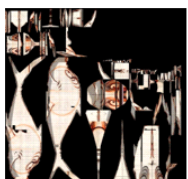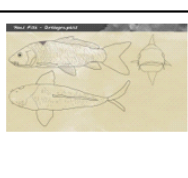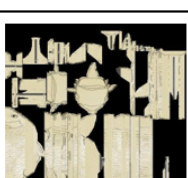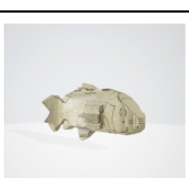
Table 1. Hard Surface Character: Ship, Yacht, JetSki, Boat, Battle Cruiser, and SkySanctuary.

Character	Resolution	File	Thresold/Noise	Input 2D Image	Output UV Map	Output 3D
Ship	936x458	png	190/33			
Yacht	1892x964	jpg	253/12			
JetSki	1350x698	jpg	190/12			
Boat	1920x1048	jpg	190/12			
Battle Cruiser	1200x1636	jpg	229/12			
SkySanctuary	1600x900	Jpg	229/12			

As a result of experimenting with 12 characters, the SketchTo3D tool visualizes various 2D characters, such as hard surfaces and organics, at a fast performance speed of less than a minute. Anyone can immediately receive the final output without 3D training. Furthermore, this advantage is necessary for the animation production pipeline, which requires a previsualization stage with simplified

Table 2. Organic Character: Diver01, Diver02, Goat, Turtle, Fish01 and Fish02.

Character	Resolution	File	Thresold/ Noise	Input 2D Image	Output UV Map	Output 3D
Diver01	1300x821	jpg	212/12			
Diver02	1920x2150	jpg	212/12			
Goat	1000x647	jpg	146/12			
Turtle	651x528	png	178/12			
Fish01	2500x1618	jpg	118/12			
Fish02	1600x974	jpg	178/12			

modeling from 2D design as soon as possible. This process with simple settings will expand the visual scope with efficient workflow.

However, in SketchTo3D tools, many functions are developed in Maya's built-in programs, which are limited to creating high-quality output. For example, a character does not connect smoothly using the Boolean intersection technique, and the shape

is not properly implemented in the blind spot. Also, the colors of the parts where each side overlaps do not match well owing to the projection map. With these problems, additional manual work seems to be done but expect to be beneficial for the educational system and previsualization stage, those who want to see it in 3D quickly.

REFERENCE

[1] T.J. Choi, H. Lee, J.H. Kim, and E.S. Kim, "Implementation of Interactive 3D Aquaium from 2D Images Using 3D Model Conversion Algorithm," *Journal of Korea Multimedia Society,* Vol. 18, No. 12, pp. 1562-1568, 2015.

[2] D.D. Hoffman, *Visual Intelligence: How We Create What We See,* WW Norton & Company, 2000.

[3] M. Dvorožňák, D. Sýkora, C. Curtis, B. Curless, O. Sorkine-Hornung, and D. Salesin, "Monster Mash: a Single-view Approach to Casual 3D Modeling and Animation," *ACM Transactions on Graphics(TOG)*, Vol. 39, No. 6, pp. 1-12, 2020.

[4] T. Igarashi, S. Matsuoka, and H. Tanaka. "Teddy: A Sketching Interface for 3D Freeform Design," *SIGGRAPH '99: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 409-416, 1999.

[5] P. Borosán, M. Jin, D. DeCarlo, Y. Gingold, and A. Nealen, "Rigmesh: Automatic Rigging for Part-Based Shape Modeling and Deformation," *ACM Transactions on Graphics(TOG)*, Vol. 31, No. 6, pp. 1-9, 2012.

[6] Z. Levi and C. Gotsman, "ArtiSketch: A System for Articulated Sketch Modeling," *Computer Graphics Forum*, Vol. 32, No. 2pt2, pp. 235-244, 2013.

[7] M. Jin, D. Gopstein, Y. Gingold, and A. Nealen, "AniMesh: Interleaved Animation, Modeling, and Editing," *ACM Transactions on Graphics*

(*TOG*), Vol. 34, No. 6, pp. 1-8, 2015.

[8] S. Saito, T. Simon, J. Saragih, and H. Joo, "PIFuHD: Multi-level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 81-90, 2020.

[9] Y. Kim, S. An, J.H. Lee, and S. Bae, "Agile 3D Sketching with Air Scaffolding," *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1-12, 2018.

[10] J. Wang, J. Lin, Q. Yu, R. Liu, Y. Chen, and S.X. Yu, "3D Shape Reconstruction from Free-Hand Sketches," *arXiv Preprint*, arXiv: 2006.09694, 2020.

[11] H. Fan, H. Su, and L.J. Guibas, "A Point Set Generation Network for 3D Object Reconstruction from a Single Image," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 605-613, 2017.

[12] ColorAtPoint, https://download.autodesk.com/global/docs/maya2013/en_us/CommandsPython/colorAtPoint.html (accessed January 10, 2022).

[13] OpenCV, https://docs.opencv.org/3.4/d3/dc0/group__imgproc__shape.html#ga95f5b48d01abc7c2e0732db24689837b (accessed April 14, 2022).

[14] Leviathans: The Great War(2022), https://www.kickstarter.com/projects/cgl/leviathans-the-great-war (accessed July 4, 2022).

[15] Alec MacLean Design, https://www.pinterest.co.kr/amacleandesign/ (accessed July 4, 2022).

[16] Deviant Art(2020), https://www.deviantart.com/shipschematicsnet/art/Battle-Cruiser-D-7-Klolode-Generation-1-856965322 (accessed July 11, 2022).

[17] Sky Sanctuary(2011), https://bryskye.tumblr.com/post/3378980181/concept-art-from-halo-reach-multiview (accessed July 11, 2022).

[18] A. Rivers, F. Durand, and T. Igarashi. "3D Modeling with Silhouettes," ACM Transac-

tions on Graphics, Vol. 29, Issue 4, Article No. 109, pp 1-8, 2010.

[19] Dreamstime(2000), https://www.dreamstime. com/diver-man-vector-animated-character- creation-set-under-water-scuba-diver-snork- eling-diving-full-length-front-side-diver- man-vector-image104668137 (accessed July 4, 2022).

[20] Sea Diver Character Design(2020), https://www. artstation.com/artwork/2qnr4e (accessed July 4, 2022).

[21] Jason Sadler(2006), https://ar.pinterest.com/ pin/832110468628973434/ (accessed July 4, 2022).

[22] P.N. Dudley, R. Bonazza, T.T. Jones, J. Wyneken, and W.P. Porter, "Leatherbacks Swimming in Silico: Modeling and Verifying Their Momentum and Heat Balance Using Computational Fluid Dynamics," *PLoS One 9,* No. 10, e110701, 2014.

[23] Jason Sadler(2022), https://www.jasonsadler art.com/ (accessed July 4, 2022).

[24] CG Art and Illustration(2014), http://vikkirh. blogspot.com/2014/10/ (accessed July 4, 2022).

[25] P.H. Mirvis, "Flow: The Psychology of Optimal Experience," *The Academy of Management Review,* Vol. 16, No. 3 pp. 636-640, 1991.

Balgum Song

currently an assistant professor at Dongseo University, Depart- ment of International College, focusing on rigging, and anima- tion pipeline. His research inter- ests include modeling, character setup, animation, VFX, cloth, fur and hair simulation.

Chul Soo Kim

currently a professor at Indus- trial Design Department, Pukyong National University. His inter- ests include Product Design, Interior Design and Public design.