Research Statement

Suzi Kim (kimsuzi@kaist.ac.kr)

Oct 2021

Throughout my Ph.D. studies, I aim to (1) extract meaningful semantics from various media, i.e., image, video, and three-dimensional (3D) model and (2) apply the semantics to different areas such as retrieval, urban modeling, film production, bio modeling, web-based application, and security (Figure 1). With the rapid development of digital media, there is a growing need for innovative media-based technologies that allow non-experts to easily create and control the various types of media. We can benefit from the automatic extraction of meaningful semantics from the media themselves and reuse the semantics to generate or manage the media. However, unveiling insightful properties of the media and designing new applications require a fine comprehension of existing graphics and vision techniques and careful reasonings from setting up the research hypotheses to drawing conclusions. My past and current research tackle such issues from different perspectives through various combinations of input media and their application (Figure 2).

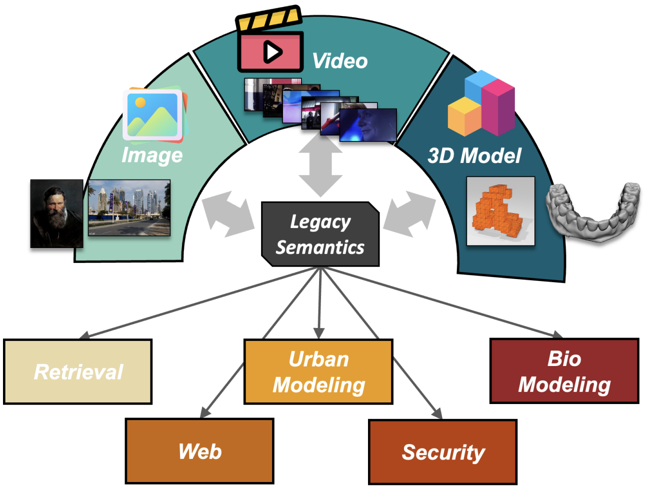

Figure 1: Research scope

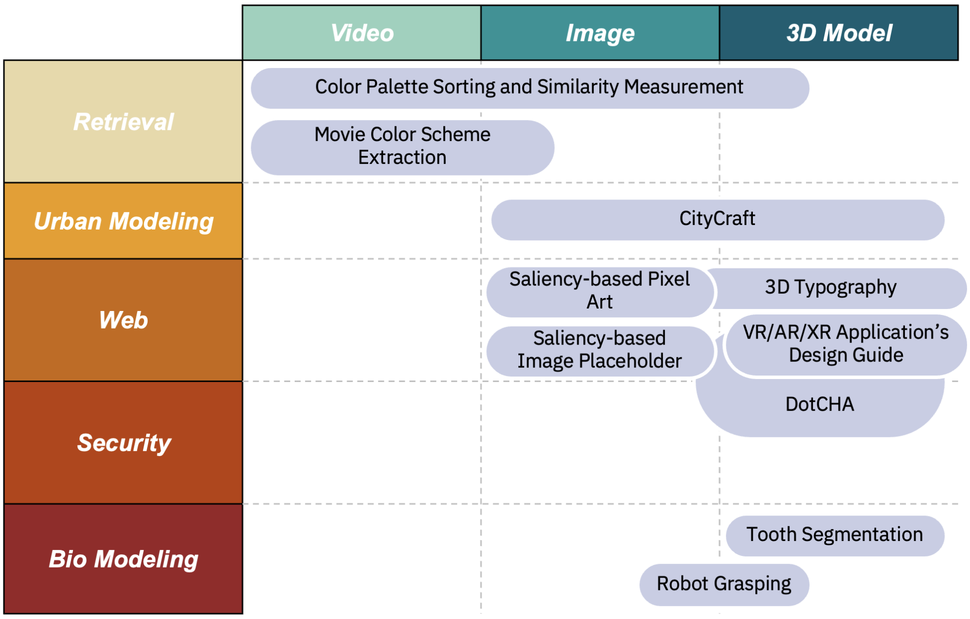

Figure 2: Research area overview

1 Past Research

CityCraft: 3D City Modeling from a Single Image

I proposed a method to generate a 3D virtual model of an imaginary city from a single street-view image to represent the appearance of the city in a given input photograph [11], as shown in Figure 3. It differs from reconstruction approaches, which generate a city model by guessing the city name from the input photograph or use multiple photos [16]. In contrast, this research combines inverse procedural modeling [1] and procedural modeling [15] to **identify where to generate the city, what to allocate in the city, and how to arrange the components**. It employed generative adversarial networks (GANs) and convolutional neural networks (CNNs) to create a terrain map and identify the components and styles that represent the virtual city appearance [3]. I demonstrated that the proposed system creates 3D virtual cities that are visually similar in terms of plausibility and naturalness to actual cities corresponding to input photographs from around the world. This is the first work to generate a city model including all general city components, including streets, buildings, and vegetation, to match the style of a single input image.

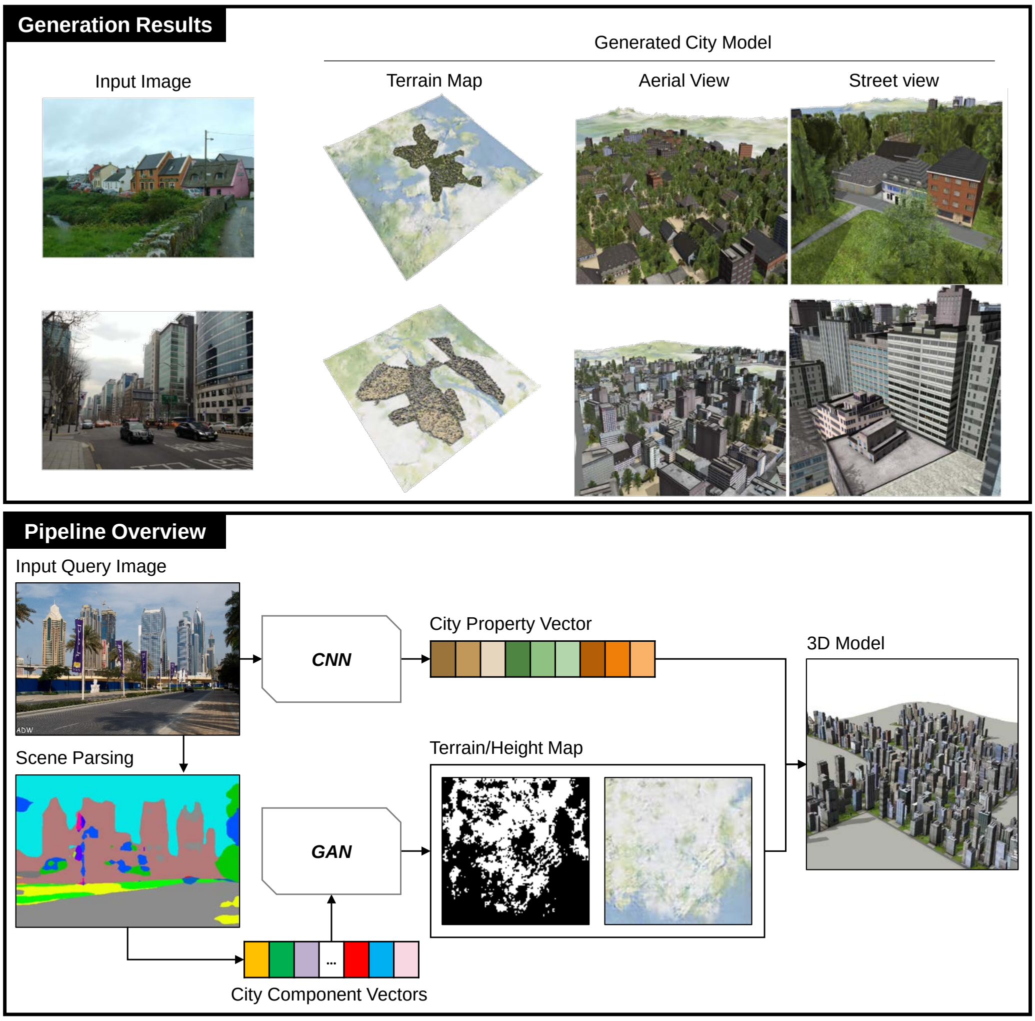

Figure 3: CityCraft generates a 3D city model which represents the appearance of the street in the given street-level photography by guessing unknown factors affecting the appearance of the city.

Color Palette Sorting and Similarity Measurement

Color palette is one of the simplest and most intuitive descriptors that can be extracted from images or videos [13]. I proposed a method to assess similarity between color palettes by sorting and aligning them [10]. Since color ordering in palettes affects visual similarity evaluation, the proposed similarity measurement aims to assess palette similarity regardless of their color order. Palettes are sorted to minimize the geometric distance between colors, and aligned to share common color tone order. Previous palette similarity measures compare only colors not considering overall palette tones. I proposed dynamic closest color warping (DCCW) to calculate the minimum distance sum between colors and the graph connecting the colors in the other palette. I

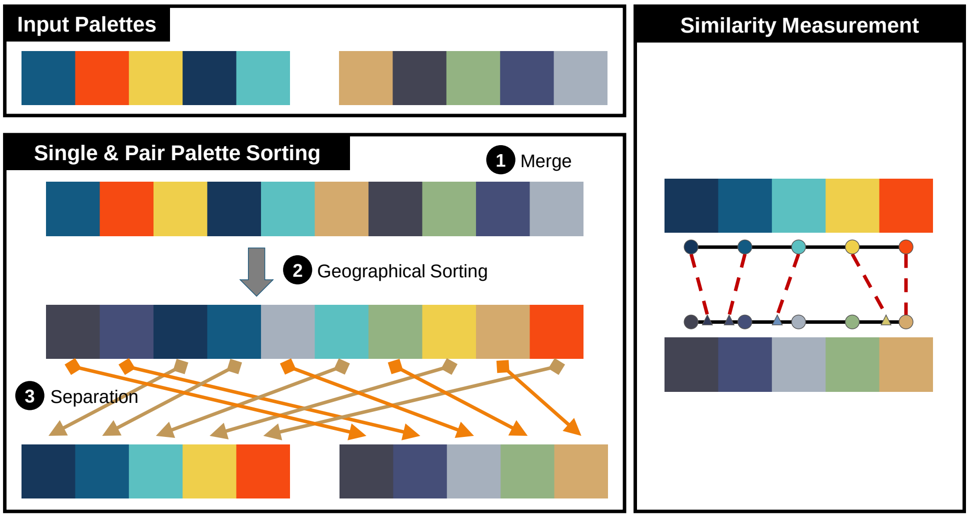

Figure 4: A method to assess similarity between color palettes by sorting and aligning them

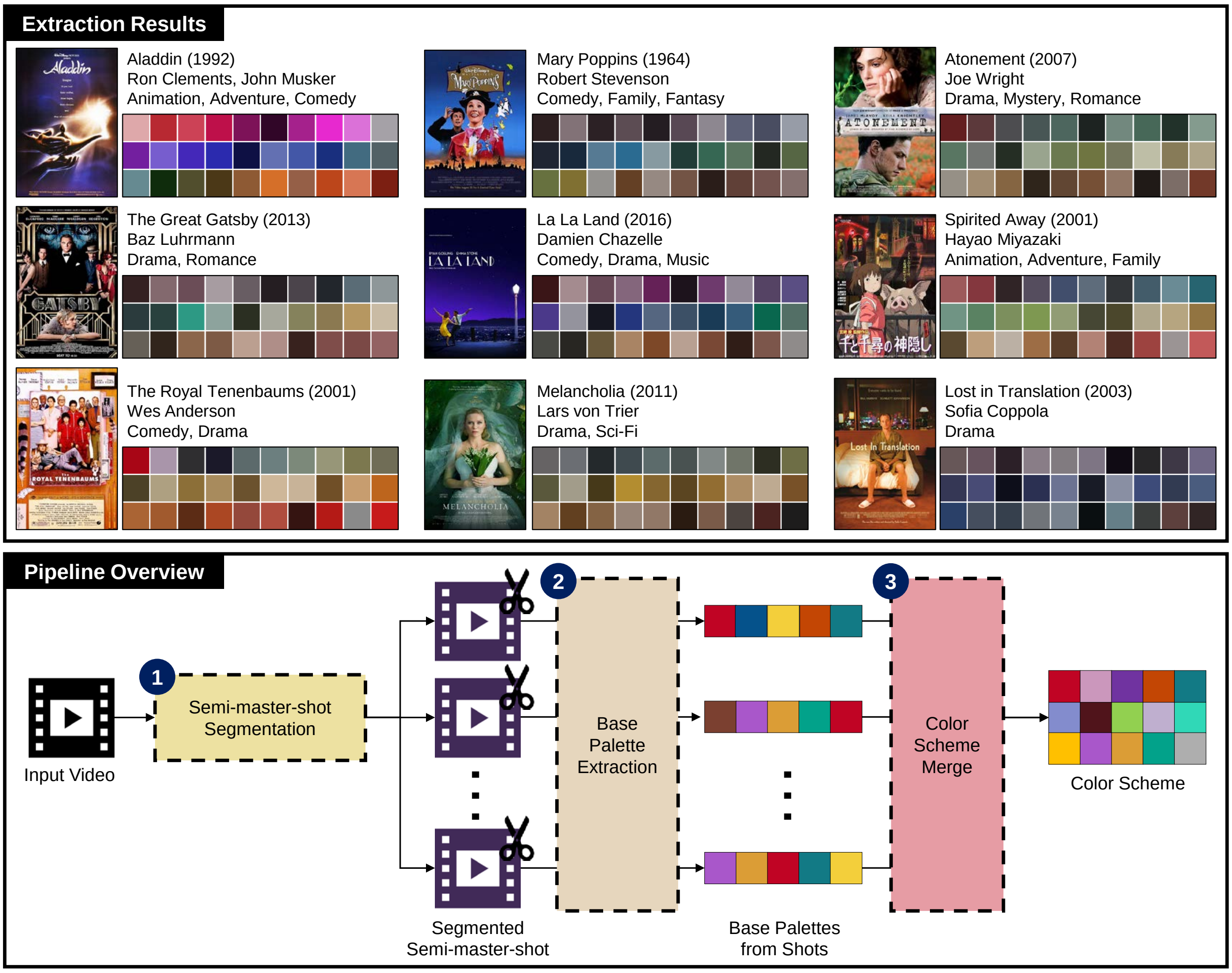

Figure 5: An automated color scheme extraction using saliency maps bottom-up from semi-master shots

evaluated the proposed palette sorting and DCCW with several datasets and demonstrated that DCCW outperforms previous methods in terms of accuracy and computing time. The prototype implementation is available at `https://suzikim.kaist.ac.kr/`.

Automatic Color Scheme Extraction from Movies

A color scheme is an association of colors, i.e., a subset of all possible colors, that represents a visual identity [17, 2]. I proposed an automated method to extract a color scheme from a movie [8], as shown in Figure 5. Since a movie is a carefully edited video with different objects and heterogeneous content embodying the director's messages and values, it is a challenging task to extract a color scheme from a movie as opposed to a general video filmed at once without distinction of shots or scenes. Despite such challenges, color scheme extraction plays a very important role in film production and application. The color scheme is an interpretation of the scenario by the cinematographer and it can convey a mood or feeling that stays with the viewer after the movie has ended. It also acts as a contributing factor to describe a film, like the metadata fields of a film such as a genre, director, and casting. Moreover, it can be automatically tagged unlike metadata, so it can be directly applied to the existing movie database without much effort. The proposed method produces a color scheme from a movie in a bottom-up manner from segmented shots. I formulated color extraction as a selection problem where perceptually important colors should be selected using saliency [12]. I introduced a semi-master-shot, an alternative unit defined as a combination of contiguous shots taken in the same place with similar colors. Using real movie videos, I demonstrated and validated the plausibility of the proposed technique. The implementation is available at `https://github.com/SuziKim/ICMR2020-MovieColorSchemer`.

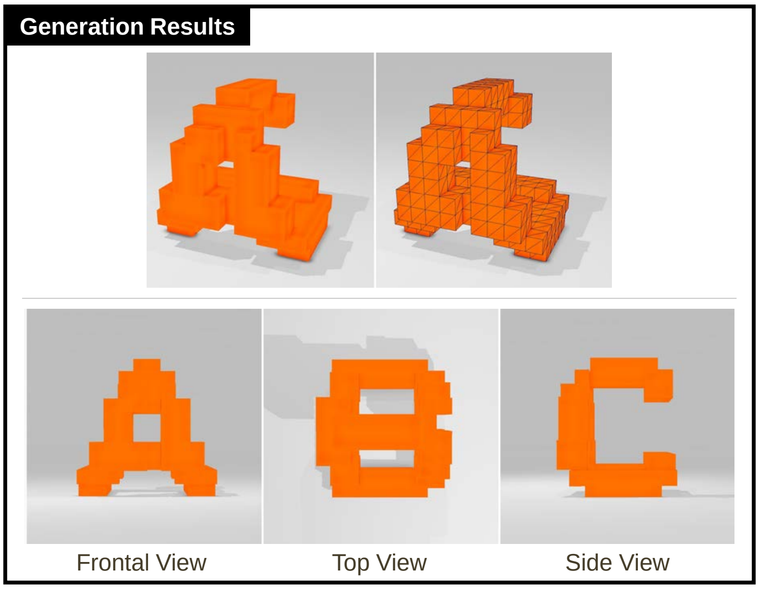

Figure 6: A method to design a 3D typography from given input letters automatically

Automatic Generation of 3D Typography

3D typography refers to the arrangement of text in 3D space. It injects vitality into the letters, thereby giving the viewer a strong impression which is hard to forget. These days, 3D typography plays an important role in daily life beyond the artistic design. It is easy to observe the 3D typography used in the 3D virtual space such as movies or games. Also, it is used frequently in signboard or furniture design. Despite its noticeable strength, most of the 3D typography is generated by just a simple extrusion of flat 2D typography. Comparing with 2D typography, 3D typography is more difficult to generate in a short time due to its high complexity. I introduced a method to design a 3D typography from given input letters automatically [4]. As shown in Figure 6, I focused on the optical illusion technique to generate an object which shows different texts depending on the viewer's perspective. I suggested a method satisfying two conditions: 1) it must create a 3D model which provides different contents in accordance with different perspectives and 2) the 3D model must have orthogonal projections that are identical with the corresponding shape of letters.

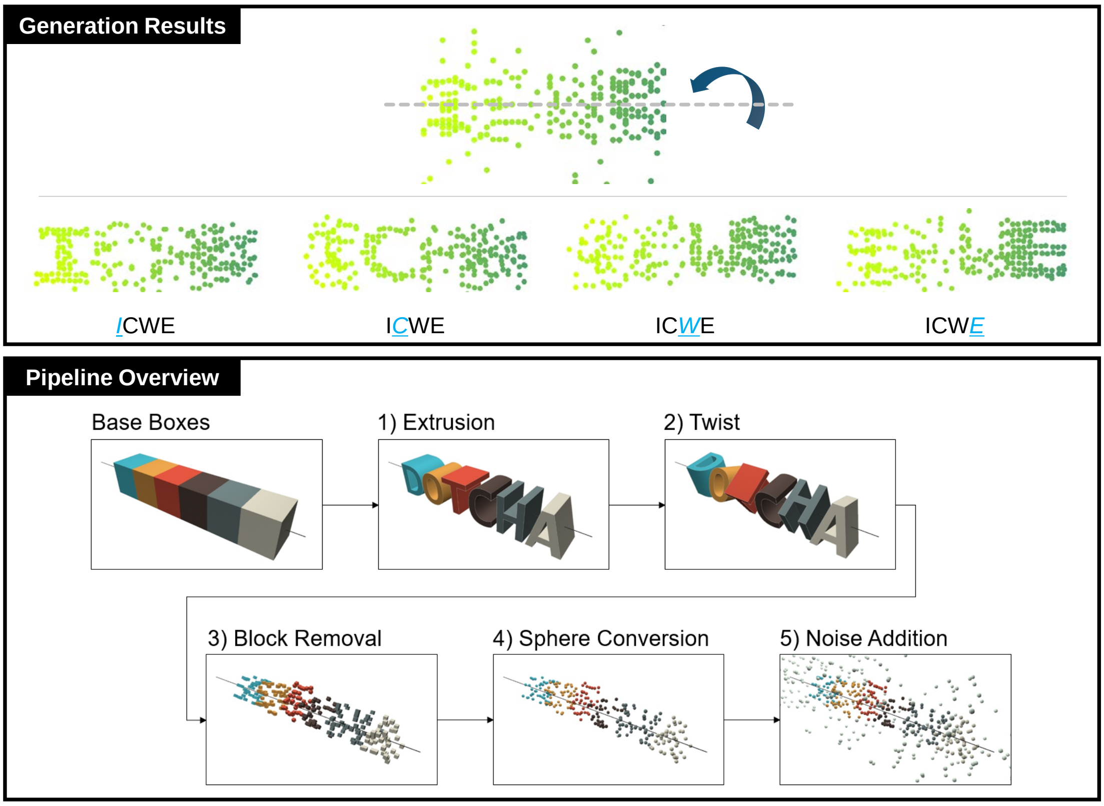

Figure 7: A new type of 3D text-based CAPTCHA to improve usability and security

DotCHA: A 3D Text-Based Scatter-Type CAPTCHA

I introduced a new type of 3D text-based CAPTCHA, called DotCHA [6, 9], which relies on human interaction and overcomes the limitations of existing 2D and 3D CAPTCHAs. As shown in Figure 7, DotCHA asks users to rotate a 3D text model to identify the correct letters. The 3D text model is a twisted form of sequential 3D letters around a center pivot axis, and it shows different letters depending on the rotation angle. Because each model consists of many small spheres instead of a solid letter model, DotCHA is classified as a scatter-type CAPTCHA and resists character segmentation attacks. Moreover, DotCHA is resistant to machine learning attacks because each letter is only identified in a particular direction. I demonstrated that DotCHA is resistant to existing types of attacks while maintaining usability. The prototype implementation is available at `https://suzikim.github.io/DotCHA/`.

Tooth Segmentation

I proposed an automatic method to separate the gingiva and individual teeth from a dental mesh [5]. As shown in Figure 8, I defined a transverse plane that produces a cross-section of tooth lingual and labial surfaces, preserving the shape of individual teeth. The upper vertices from the transverse plane, which belong to the tooth, are projected onto the transverse plane and partitioned into individual teeth. I applied region growing to the remaining non-segmented parts to determine the cluster the vertices belong to, and the proposed approach is fully automatic, i.e., segmentation does not require user interaction for feature point search or tooth boundary markers.

XR/AR/VR Application's Design Guide

We have become accustomed to flat screens over a long time because flat screens are both easy to manufacture and easy to carry around. However, there is a huge gap between the 3D experiences that we see directly with our eyes and those viewed through a flat screen. I compared 3D experiences through eyewear and other media [7]. Based on the strengths of eyewear, I suggested four design considerations for eyewear applications vividly conveying the actual 3D experience: (1) less modeling labor, (2) easy manipulation and interaction, (3) no need for extra action except gazing, and (4) complete immersion in the virtual world, as shown in Figure 9.

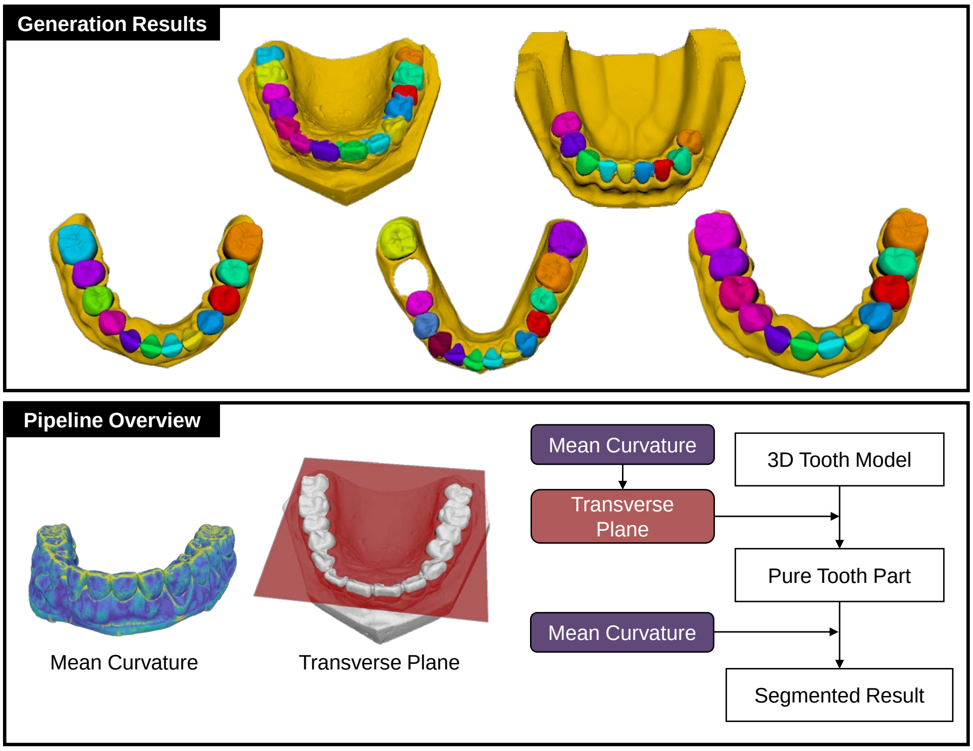

Figure 8: An automatic method to separate the gingiva and individual teeth from a dental mesh without knowing the number of teeth

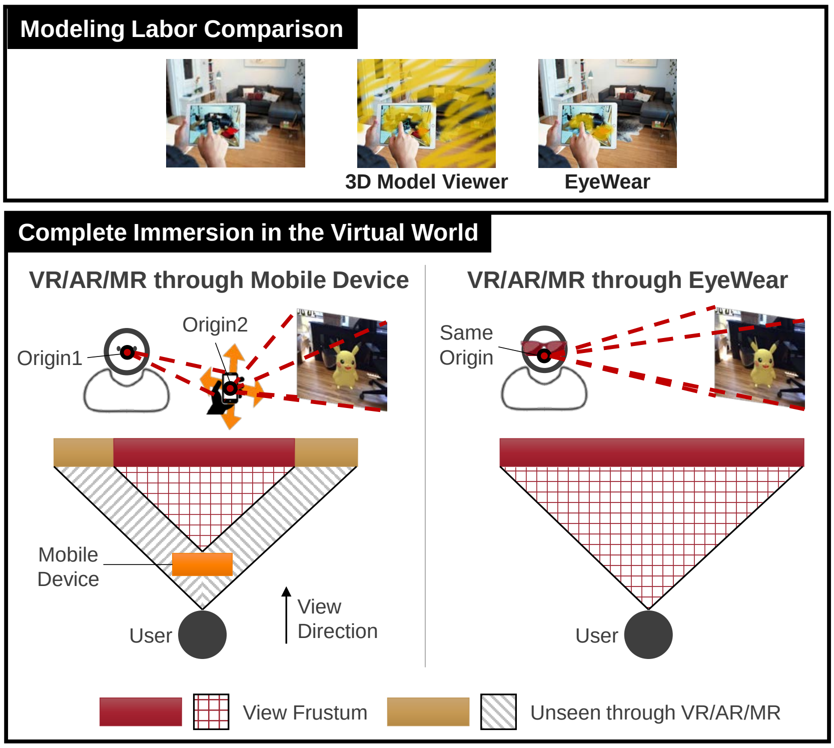

Figure 9: Comparison of 3D experiences through eyewear and other media and suggest design guide for eyewear applications

2 Ongoing and Future Research

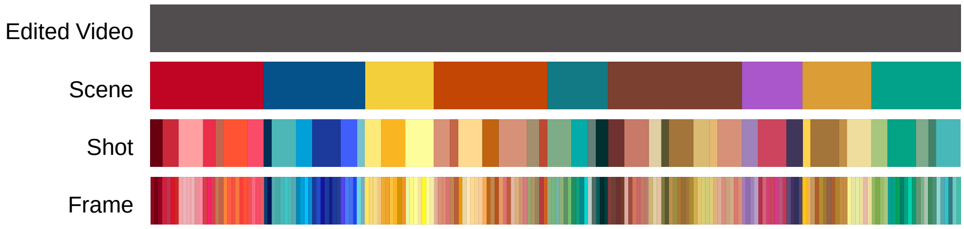

Figure 10: Typical structure of edited video

Scene Segmentation from the Edited Video

As shown in Figure 10, an edited video consists of a linear sequence of scenes, where each scene consists of several shots. A scene is a sequence of interrelated shots that share a common semantic thread, whereas a shot is a sequence of frames filmed by a single camera without interruption. Shot and scene are the basic units for decomposing the edited video to manage and control the contents. Hence, segmentation of shot and scene is an essential technique for manipulation and retrieval of edited videos.

Shot changes can occur abruptly or through a gradual transition. Abrupt transitions occur over a single frame due to camera switch; whereas gradual transitions, such as dissolve, fade-in, fade-out, and wipe, stretch over several frames with a variety of video effects. Shot segmentation aims to detect these transitions by grouping frames using image similarity. Scene segmentation is more challenging than shot segmentation because the scenes are segmented according to the semantic context. Cinematography and editing techniques have developed, the boundary between scenes has become ambiguous as more complex than simple techniques such as dissolve and overlap.

Although most shot segmentation transitions are detected by video features alone, because the shot is filmed as a single take, scene segmentation methods generally use multimodal features to reduce ambiguity, i.e., video, audio, and text [14]. However, segmentation still suffers from poor accuracy, so a different approach is needed to break away from the existing multimodal methods. So, I plan to adopt the movie script as ground truth and minimize the number of features to reduce the time costs.

Also, I plan to define and extract a new concept of shot and scene. For certain specific purposes, precise segmentation may not be necessary, I will design a new concept of segmentation result that is consistent with the objective. For example, in the previous research of automatic color scheme extraction [8], I defined a semi-master-shot which combines contiguous shots taken in the same location with similar colors. The semi-master-shot aims to contain all the characters, representing the atmosphere of all the space being filmed.

3D Scene Reconstruction from the Edited Video

The aforementioned scene segmentation can be a fundamental technique in various video manipulation fields, such as video coloring and video to text conversion. I plan to adopt the segmentation to reconstruct a 3D scene from discontinuous scenes of edited video. 3D scene reconstruction refers to the capturing and rebuilding of a 3D model from a given image or video. Most 3D scene reconstruction is based on a RGB image or non-edited single-shot video.

If scenes are filmed discontinuously in an edited video like a movie, there are two challenging issues to solve. First, not all spaces are filmed slowly and stably, so additional inferring techniques are required to resolve blur and occlusions. Second, even if the two scenes are filmed in the same place, there may be a discrepancy of contents that do not match perfectly because it may have been taken at different times.

In recent years, online video-sharing platforms have emerged all over the world and people have started to shoot and edit their own videos. The shift from static images and single shots video to edited videos is inevitable, and technology should be able to support the manipulation of edited videos. 3D scene reconstruction reduces the production costs in the film and game industries by providing a relatively fast automatic 3D scene generation with minimal effort by non-experts. It helps to recreate the space of the past era and rebuild the production design in the edited video that has already been filmed.

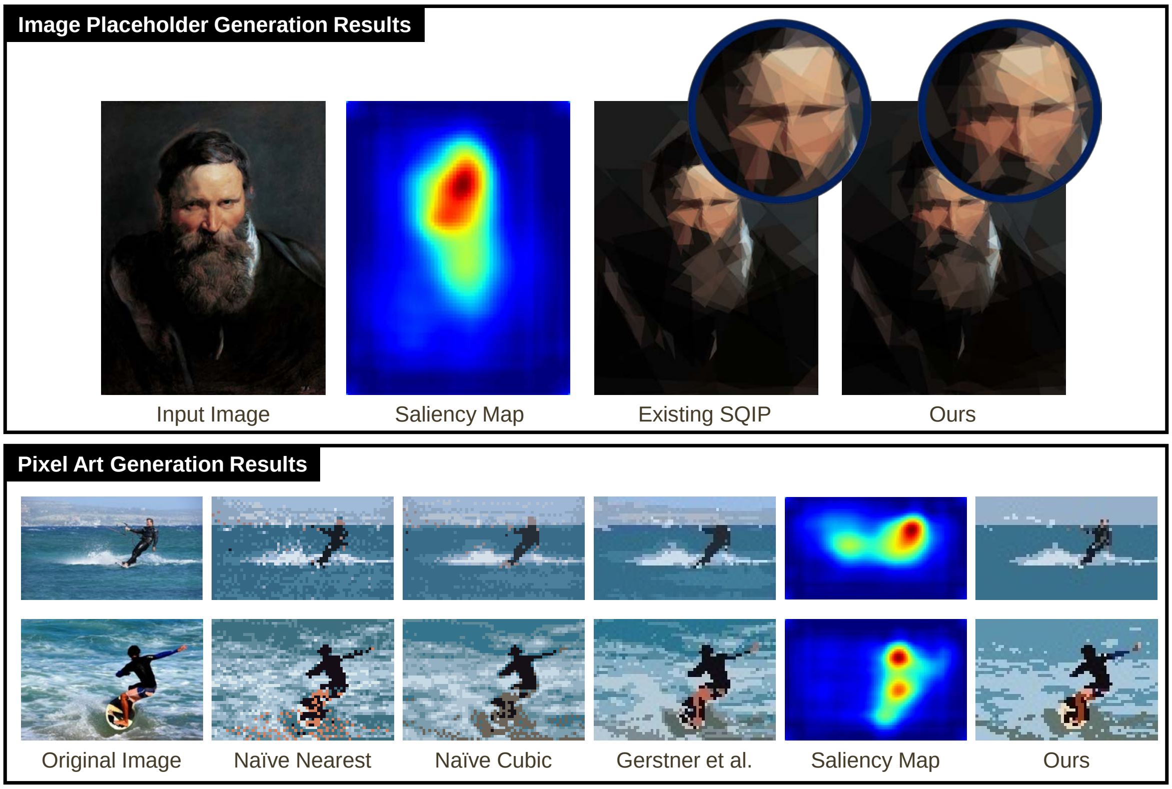

Figure 11: A saliency-based (1) X3D image placeholder generation and (2) pixel art image to increase the detail while preserving the file size of results

Saliency-based Image Processing

I plan to propose two saliency-based image processing methods to generate (1) an X3D-based image place holder and (2) a pixel art image. X3D-based image placeholder is a small file-size X3D-based image used instead of original images for fast and efficient loading of large file-size images or large-scale images on the web. Pixel art is an abstraction of high-resolution images into very low-resolution images with reduced color palettes. However, both techniques increase the file size inevitably for a detailed description. Saliency map can be applied in both techniques to produce high level-of-detail without increasing file size. Saliency maps represent pixel significance within each image by following human fixation points. They encourage increasing the detailed description of the higher saliency part in the original image by reducing the details of the lower saliency part to keep the file size.

References

[1] Sawsan AlHalawani, Yong-Liang Yang, Peter Wonka, and Niloy J. Mitra. What makes london work like london? *Comput. Graph. Forum*, 33(5):157–165, 2014.

[2] Huiwen Chang, Ohad Fried, Yiming Liu, Stephen DiVerdi, and Adam Finkelstein. Palette-based photo recoloring. *ACM Trans. Graph.*, 34(4):139:1–139:11, 2015.

[3] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei A. Efros. What makes paris look like paris? *ACM Trans. Graph.*, 31(4):101:1–101:9, 2012.

[4] Suzi Kim and Sunghee Choi. Automatic generation of 3d typography. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference, SIGGRAPH '16, Anaheim, CA, USA, July 24-28, 2016, Posters Proceedings*, pages 21:1–21:2. ACM, 2016.

[5] Suzi Kim and Sunghee Choi. Automatic tooth segmentation of dental mesh using a transverse plane. In *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2018, Honolulu, HI, USA, July 18-21, 2018*, pages 4122–4125. IEEE, 2018.

[6] Suzi Kim and Sunghee Choi. Dotcha: A 3d text-based scatter-type CAPTCHA. In Maxim Bakaev, Flavius Frasincar, and In-Young Ko, editors, *Web Engineering - 19th International Conference, ICWE 2019, Daejeon,*

South Korea, June 11-14, 2019, Proceedings, volume 11496 of *Lecture Notes in Computer Science*, pages 238–252. Springer, 2019.

[7] Suzi Kim and Sunghee Choi. The hitchhiker's guide to the eyewear applications. In Robert Harle, Katayoun Farrahi, and Nicholas Lane, editors, *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers, UbiComp/ISWC 2019 Adjunct, London, UK, September 9-13, 2019*, pages 633–636. ACM, 2019.

[8] Suzi Kim and Sunghee Choi. Automatic color scheme extraction from movies. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, ICMR '20, page 154–163. Association for Computing Machinery, 2020.

[9] Suzi Kim and Sunghee Choi. Dotcha: An interactive 3d text-based captcha. *Journal of Web Engineering*, pages 837–864, 2020.

[10] Suzi Kim and Sunghee Choi. Dynamic closest color warping to sort and compare palettes. *ACM Trans. Graph.*, 40(4):95:1–95:15, 2021.

[11] Suzi Kim, Dodam Kim, and Sunghee Choi. Citycraft: 3d virtual city creation from a single image. *Vis. Comput.*, 36(5):911–924, 2020.

[12] Junting Pan, Elisa Sayrol, Xavier Giró-i-Nieto, Kevin McGuinness, and Noel E. O'Connor. Shallow and deep convolutional networks for saliency prediction. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 598–606. IEEE Computer Society, 2016.

[13] Huy Q. Phan, Hongbo Fu, and Antoni B. Chan. Color orchestra: Ordering color palettes for interpolation and prediction. *IEEE Trans. Vis. Comput. Graph.*, 24(6):1942–1955, 2018.

[14] Daniel Rotman, Dror Porat, Gal Ashour, and Udi Barzelay. Optimally grouped deep features using normalized cost for video scene detection. In Kiyoharu Aizawa, Michael S. Lew, and Shin'ichi Satoh, editors, *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval, ICMR 2018, Yokohama, Japan, June 11-14, 2018*, pages 187–195. ACM, 2018.

[15] Ruben Michaël Smelik, Tim Tutenel, Rafael Bidarra, and Bedrich Benes. A survey on procedural modelling for virtual worlds. *Comput. Graph. Forum*, 33(6):31–50, 2014.

[16] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.*, 25(3):835–846, 2006.

[17] Jianchao Tan, Jose I. Echevarria, and Yotam I. Gingold. Efficient palette-based decomposition and recoloring of images via rgbxy-space geometry. *ACM Trans. Graph.*, 37(6):262:1–262:10, 2018.