Napkinmatic 3D

Yosun Chang
yc@AIMagical.com
AIMagical Permute.xyz
San Francisco, CA, USA

ABSTRACT

We present a novel approach for 3D content creation via a platform that connects reality to AI, on paper.

CCS CONCEPTS

• Human-centered computing → Interaction paradigms; **Ubiquitous and mobile computing**; **• Computing methodologies → Artificial intelligence**; **Mixed / augmented reality**.

KEYWORDS

augmented reality, AI, 3D creation, ubiquitous computing

ACM Reference Format:
Yosun Chang. 2023. Napkinmatic 3D. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Real-Time Live! (SIGGRAPH '23 Real-Time Live!), August 06-10, 2023.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3588430.3597253

1 INTRODUCTION

In our previous Napkinmatic (2D) paper [Chang 2023a], we showcase the concept of a generalized reality user interface sometimes called spatial computing interface, connecting paper with AI (REST endpoints). Here, we expound on the 3D version of the app.

2 PARADIGM TURNING A 2D NAPKIN SKETCH INTO A 3D CREATION

Adding in an extra dimension runs into the information theory problem of additional complexity increasing with added degrees of freedom. Simply attempting brute-force grouping of tools around semantic concepts result in the "armageddon" known as traditional UI for 3D creation software, as exemplified in industry software such as Maya or Blender or 3DS Max.

2.1 Archetypes and Intuitive Hybrid UI

We utilize archetypes that are intuitive using hybrid interfaces that connect realities. We also use AI and contextual interfaces to hone in on the most plausible operations the user may wish to perform. The system is extensible with a plugins system, where anyone can plug in their API, with also an App Store format.

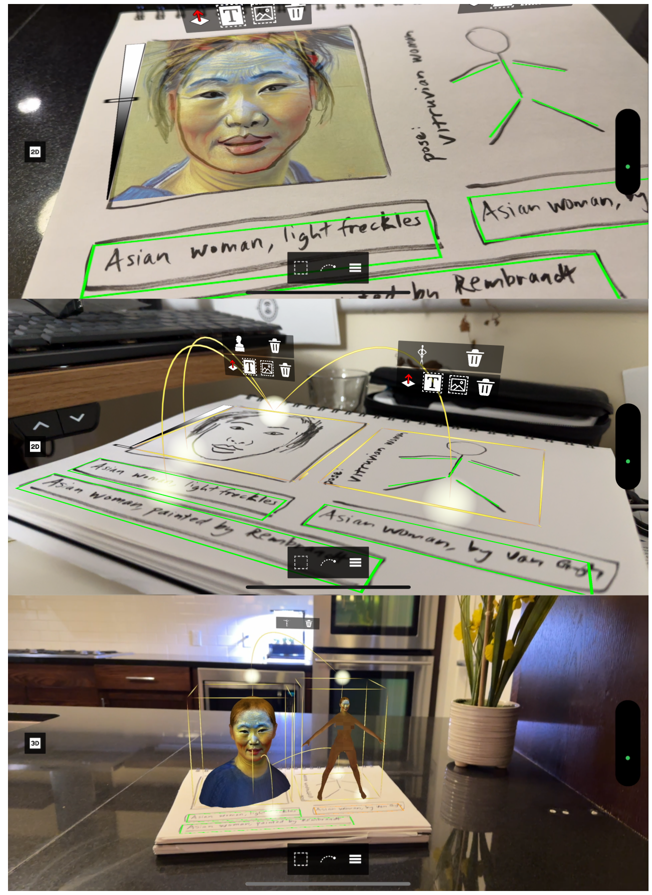

Figure 1: Screenshots (top to bottom): Light Trajectories connect modules, relevant operations $\hat{\Theta}$ based on objects $\mathbb{O}$

2.2 Implementation

Effectively, objects are recognized and intuitive nodal graph connectors create relations among objects. Each object is identified to help deduce the most plausible operations in the contextual semantic menu. The following section abstractifies this concept.

3 "NAPKINSPACE3D" OR $\mathbb{N}^{3d}$

Napkinspace $\mathbb{N}$ consists of the visual element objects (objectspace) $\mathbb{O}$ and operators Θ and connective relations when an operator is applied, hence $\{\mathbb{O}^{2d}, \mathbb{O}^{3d}, \Theta\} \subset \mathbb{N}$ in both 2D napkinspace and the thus inferred 3D napkinspace.

Objects are identified by bounding boxes, in both 2D and 3D. We can slice reality using the "Light Scalpel" tool to draw bounding boxes, that can also be extruded from 2D to 3D (and back again),

ACM ISBN 979-8-4007-0158-0/23/08.
https://doi.org/10.1145/3588430.3597253

to load semantically relevant operators for each space. Objects can be connected to each other using the "Light Trajectory" tool. Connected objects can have operators applied.

Operators are denoted with the hat symbol, and they transform objects. They are not necessarily 1:1 and may be any combination of inputs and outputs of $\mathbb{I}$ image, $\mathbb{C}$ text, $\mathbb{S}$ demarcated sections, and other objects in Napkinspace $\mathbb{N}$.

$$\hat{\Theta}(\mathbb{I}, \mathbb{C}, \mathbb{M}, \mathbb{S}...) \mapsto (\mathbb{I}, \mathbb{C}, \mathbb{M}, \mathbb{S}...) \quad (1)$$

For example, recognized text or images in $\mathbb{O}^{2d}$ or 3D mesh objects in $\mathbb{O}^{3d}$ can have different operations $\hat{\Theta}$ applied to turn them into other objects in objectspace. Here are some common operators:

- Define $\hat{\mathbb{F}} = 2d \mapsto 3d$ as a face operator that transforms a face image to a 3D avatar face mesh
- Define $\hat{\mathbb{S}} = 2d \mapsto 2d$ as a shirt operator (image to 2D UV space mapping on a 3D object), that places the $\mathbb{O}^{2d}$ 2D image in a specific part of the 3D shirt.
- Define $\hat{\mathbb{B}} = \mathbb{C} \mapsto \mathbb{C}$ as a non-dimensional (imaginary space) intermediary operator that manifests as a button, that inputs text to text of another operator, such as directional notes guiding a particular output from $\hat{\mathbb{C}}$ operator Controlnet [Zhang and Agrawala 2023] $\hat{\mathbb{B}}(\mathbb{C}) \Rightarrow \hat{\mathbb{C}}(\mathbb{C}, \mathbb{I} \mapsto \mathbb{I})$, where text $\mathbb{C}$ and image $\mathbb{I}$ are transformed to image or $\hat{\mathbb{T}}$ operator TEXTure [Richardson et al. 2023] $\hat{\mathbb{B}}(\mathbb{C}) \Rightarrow \hat{\mathbb{T}}(\mathbb{C}, \mathbb{M} \mapsto (\mathbb{I}, \mathbb{M}))$, where 3D mesh $\mathbb{M}$ and text $\mathbb{C}$ are transformed to textured $\mathbb{I}$ 3D mesh $\mathbb{M}$.

3.1 Local vs Remote Operators

Abstractifying a demarcation between client-side mobile and server-side operations, we can segregate operators into server-side remote operators vs local operators. For example, $\hat{\mathbb{C}}$ Controlnet and $\hat{\mathbb{T}}$ TEXTure are both remote server-side operators as of this writing, such compute being impractical on current generation mobile devices. Buttons $\hat{\mathbb{B}}$ are local operators, but the OCR conversion can be either local or remote, depending on the method used to convert image to text.

4 (REST) API ENDPOINTS AS OPERATORS

When API endpoints are used as operators, because the operation may be asynchronous, we utilize data structure that denote the processing state of the operation with Napkinspace object operation delayed until (status == success), as defined in [Chang 2023b].

5 SKETCH TO 3D MODEL (MAY 2023)

5.1 Simple Mesh Operations on Image

We can extract the edges of an image and then apply an array of traditional mesh creation algorithms, from extrusion to lathing and more. (These transformation matrices are reasonable and well behaved, so this is typically done locally.)

5.2 Front face profile to 3D avatar mesh

Face "AR" techniques have become commodity, and turning a sketch into a clean face photo (stable diffusion controlnet) into a deepfake face mesh or a 3D avatar mesh is straight forward. At previous SIGGRAPHS, we showed techniques for mapping what you draw on paper to various 3d meshes include faces and even origami [Chang 2019] [Chang and Grandhi 2019]. Also: 2D image to 3D digital human techniques using dlib [Chang 2021] and 3DMM [Lei et al. 2023].

5.3 Side Profile as scalar to "EigenFace"

We can take the various extrema of any curve and map it into a side profile forehead-eye-nose-chin curve, and then apply these as deformations to an "eigenface" 3D mesh.

5.4 Several images to 3D mesh

We can use techniques such as Sparsefusion [Zhou and Tulsiani 2023] and a profusion of other diffusion-based to 3D mesh [Guo et al. 2023]

5.5 Image to 3D via Shape-E

We can use the process of $\hat{C}(\mathbb{I}) = \mathbb{I} \Rightarrow \hat{T}(\mathbb{I}) = \mathbb{M}$ converting a sketch to a Pixar-like 3D model using stable diffusion controlnet, and feed it to Shap-E [Jun and Nichol 2023] Image to 3D, which runs in about 10 seconds on A10G.

Shap-E by default gives results that are not usable practically without refinements. One pipeline is to convert the Shap-E generated mesh into a stylized low poly 3D mesh, as the author did with a spinoff iOS/Android app called PolyMagical [Chang 2023c] in her SPEEDRUN "build 5 novel 3D AI apps in 5 weeks" with AI experiment. Also, re-UV and re-texturing with TEXTure is another method, as described in the next section.

6 RETEXTURING FROM PROMPT

We can re-texture our generated mesh from our handmade "prompt buttons" using TEXTure [Richardson et al. 2023].

7 CONCLUSION

We hope to have inspired you rethink everything you know about UI/UX for 3D content creation in the age of daily SOTA in AI.

REFERENCES

Yosun Chang. 2019. *SIGGRAPH '19: ACM SIGGRAPH 2019 Appy Hour* (July 2019). https://dl.acm.org/doi/10.1145/3305365.3329730

Yosun Chang. 2021. *faced.io TinyHumans simplified image to 3D digital human API.* Retrieved June 9, 2023 from https://devpost.com/software/faced-io-tinyhumans

Yosun Chang. 2023a. 50, 1 (Aug. 2023). https://doi.org/10.1145/3588427.3595357

Yosun Chang. 2023b. *Napkinmatic - Open Framework.* Retrieved May 23, 2023 from https://github.com/yosun/napkinmatic

Yosun Chang. 2023c. *PolyMagical - prompt to 3D low poly mesh.* Retrieved June 9, 2023 from https://PolyMagical.com

Yosun Chang and Uttam Grandhi. 2019. *SIGGRAPH '19: ACM SIGGRAPH 2019 Appy Hour* (July 2019). https://dl.acm.org/doi/10.1145/3305365.3329729

Yuan-Chen Guo, Ying-Tian Liu, Chen Wang, Zi-Xin Zou, Guan Luo, Chia-Hao Chen, Yan-Pei Cao, and Song-Hai Zhang. 2023. threestudio: A unified framework for 3D content generation. https://github.com/threestudio-project/threestudio.

Heewoo Jun and Alex Nichol. 2023. Shap-E: Generating Conditional 3D Implicit Functions. arXiv:2305.02463 [cs.CV]

Biwen Lei, Jianqiang Ren, Mengyang Feng, Miaomiao Cui, and Xuansong Xie. 2023. A Hierarchical Representation Network for Accurate and Detailed Face Reconstruction from In-The-Wild Images. arXiv:2302.14434 [cs.CV]

Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. 2023. TEXTure: Text-Guided Texturing of 3D Shapes. arXiv:2302.01721 [cs.CV]

Lvmin Zhang and Maneesh Agrawala. 2023. Adding Conditional Control to Text-to-Image Diffusion Models. arXiv:2302.05543 [cs.CV]

Zhizhuo Zhou and Shubham Tulsiani. 2023. SparseFusion: Distilling View-conditioned Diffusion for 3D Reconstruction. arXiv:2212.00792 [cs.CV]