ShapeShifters: Editing 3D Data with Language

Group name: Shape Shifters
Team members: Laura Roettges (roettges@wisc.edu), Abhinav Narayan Harish (aharish2@wisc.edu), Kevin Macauley (kcmacauley@wisc.edu), Keshav Sharan Pachipala (pachipala@wisc.edu)

Motivation

Utilizing tools like 3D scanners and photogrammetry software greatly facilitates the reverse engineering process, enabling the replication or rapid prototyping of objects through 3D printing, benefiting fields such as product design and architectural modeling. However, these tools are often not without some level of inaccuracy, generating fragmented point clouds due to noise or device calibration issues. The subsequent manual cleanup in software platforms like • SolidWorks®, Onshape®, Blender®, etc. demands considerable time and a steep learning curve for users to master the intricacies of these tools.

In recent years, there has been interest in utilizing language—an almost universally understood tool—to generate e.g. [1], segment e.g. [2], and manipulate e.g. [3] 3D models. We are interested in better understanding the application of 3d object manipulation to improve the precision of point clouds and streamlining the design process for creating diverse functional object variations tailored to a wide range of design applications. Specifically, we plan to explore utilizing the ShapeTalk dataset of “discriminative utterances produced by contrasting the shapes of common 3D objects” [3] and the ChangeIt3D framework which utilizes a “3D generative model of shapes” [3] to modify 3D point clouds based on the description a user provides on how to deform the model and analyze how it performs on broken point clouds due to noise or imperfect calibration.

Background:

Over the last few-years, the deep-learning community has witnessed an increasing interest in connecting different modalities of representation to increase the richness of our interaction with the world.

CLIP [4] was one of the first methods to connect language and images by proposing a common embedding space for the two modalities. Subsequent works like DALL-E [5] extend ideas from CLIP to apply it to generating and modifying images based on a text prompt.

A parallel trajectory of research has emerged in the realm of three-dimensional (3-D) modeling, exemplified by projects like CLIP-Forge [6], Dream-Fields [7], and Shape-Crafter [8], which built text conditioned 3-D generation models. However, these endeavors have primarily focused on

the direct generation of 3-D shapes, they have not placed significant emphasis on nuanced shape editing, particularly regarding specific components of the shapes. Although some efforts have been made to establish connections between language and the structural features of 3-D shapes, leveraging localization techniques for direct shape editing remains an underexplored avenue.

In this project, we explore a way of editing 3-D shapes using a multi-modal model to propose edits to the 3-D shapes. Similar to multi-modals like CLIP, we propose an edit in the latent space and utilize a shape-decoder to recover the shape in the original 3-D space.

Datasets

We plan on using the dataset provided by the authors of ShapeTalk as described here. The dataset tuples of distractor and target 3d models with utterances describing shape differences as seen in Figure 1.

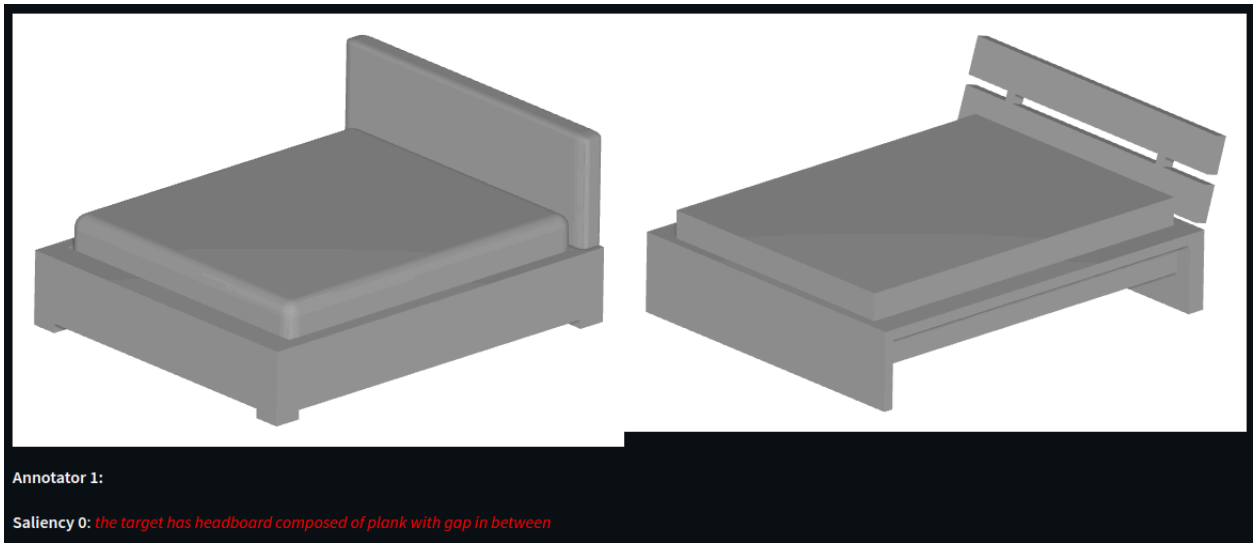

Figure 1: *An example instance of the distractor (left), target (right) and utterance (red text). (**Achlioptas, 2023**)*

The shapes themselves are used to train a latent-shape network. The latent shapes from the latent-shape network and the utterances are then used to train a latent-based neural-listener to learn the differences between the shapes based on the utterance. These two networks are then frozen and used in the final shape editing network. This dataset provides discriminative utterances for 36,391 shapes, across 30 object classes. We will sample a distribution of 6,000 shapes from this dataset to train our own model.

Additionally we aim to use 3D scanners from the UW Makerspace to acquire 3D data from the real world in order to further test the method's application in the real world.

Method

The workflow of our 2-stage pipeline is shown in Figure 2 below. Broadly, this is learnt in two stages. First we learn a shape encoder in order to represent a 3D shape into a compressed latent space and a multimodal discriminator which predicts the probability of each of the shapes matching the query of interest. In the second stage we utilize the pre-trained shape-encoder and discriminator to propose edits to the 3-D-shape based on the text query. This is done by the shape-editor module which proposes a “change” vector which is added to the shape-features before recovering the edited shape using the decoder. The frozen neural listener module supervises the second stage using the probabilities of the shapes corresponding to the text-query.

ShapeTalk’s Structure is shown below in Figure 2.

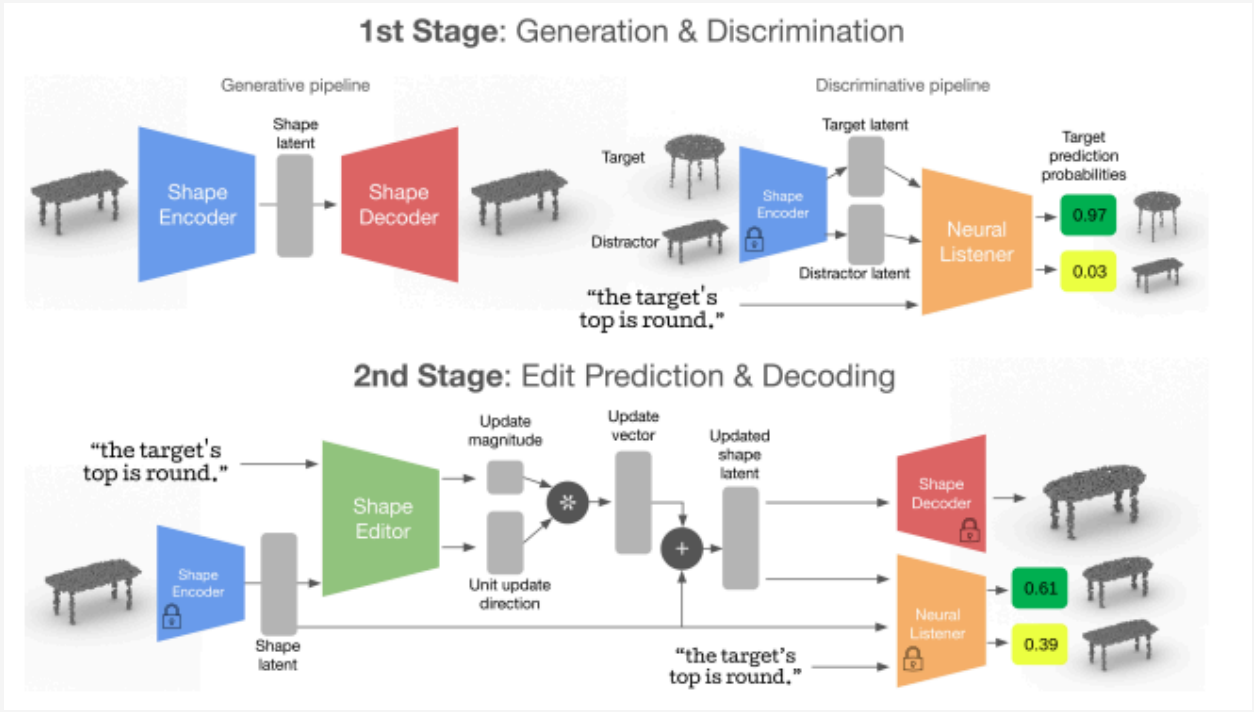

Figure 2: *Shows the framework of ChangeIt3D, the model presented in ShapeTalk* ***(Achlioptas, 2023)***

Experiment

We propose starting by downsampling the data through picking a limited subset of 6,000 shapes across 5 object classes. We choose an 85%, 5%, 10% split of the subset into train, validation, and test sets will be used to train our model. We will train ChangeIt3D for approximately 4-5 hours and assess model performance on these three distributions. In addition, we plan on utilizing carefully engineered prompts to fix the scanned real world objects with fragmented point clouds. Further, a quantitative evaluation will be conducted on reconstruction accuracy by artificially creating holes in point clouds and comparing the reconstruction with ground truth models of the same object.

We intend to evaluate the sensitivity and bias of the neural listener towards certain words in the input prompts. For this, we conduct a qualitative analysis, utilizing prompts with the same meaning to modify the same 3D shape. For instance, instead of saying the “target top is round”, we can replace it with “target top has a circular shape” and demonstrate visual results of the model’s prediction. We would also like to assess the ability of ChangeIt3D on discriminator with varying accuracy of identifying shape of interest from the provided text prompt.

For our final experiment, we plan to artificially modify our shape point cloud by introducing noise to the vertices of the 3D point cloud and assessing the strength of both our autoencoder (measured using earth mover distance) and our discriminator (assessed by accuracy).

Evaluation Metrics and Desidered Data

Since we plan to evaluate the performance of the model on real world scans with ground truths and point clouds from synthetic/ internet-sourced datasets with no available ground truth, we plan on utilizing a combination of the following evaluation metrics.

Given the interpretive nature of processing the prompts by the neural listener, we require intuitive evaluation metrics that can handle all the possible minute ambiguities that are bound to arise. For this we will use the evaluation parameters suggested in earlier works [3].

Linguistic Association Boost. When the performed change brings in the intended change as desired by the input query, the output model is expected to have a higher visio-linguistic association with the said query when compared to the input point cloud. For this a trained neural listening network that finds the described target shape from the input query and can measure the association between the input and the generated outputs with the input query will be utilized.

***localized*-Geometric Difference.** An object that is the most similar to the input image post modification is preferred. As the modifications will result in greater difference a geometric difference excluding the modified parts is preferred. A neural listener will be used to semantically segment the input and output models using the set of linguistic instructions given and after deleting the identified parts, a simple geometric difference is computed.

Class Membership. The output generated is expected to belong to the same class as the input model and the output with the higher probability to be a member of that class will be considered as the better result. We use an object classifier and utilize cross entropy loss on the predicted probabilities.

Reconstruction Loss. For the real world scans with broken point clouds, the geometric distance (Earth Mover Distance) between the ground truth and the uncorrupted point cloud would be used.

EMD computes the amount of work needed to transform a one point cloud to another. Being highly sensitive to the distribution/density of the points in the generated model, this gives us a better measure of similarity that is more in line with human visual judgment.

Timetable

Milestone	Due Date	Subtasks	Start Date	Off Track Date	Completed?	Comments
Submit Project Proposal	**23-Feb**	Make group	1-Feb	12-Feb	Yes	
		Research papers for inspiration	12-Feb	20-Feb	Yes	
		Decide on Topic	19-Feb	21-Feb	Yes	
Install and Validate ShapeTalk model using pretrained weights and networks	**15-Mar**	Ensure we have the hardware/resources to effectively run the neural model	23-Feb	28-Feb		
		Delegate discrete steps and define meeting times	23-Mar	28-Feb		
		Review of github code base and parameters for manipulation	23-Feb	1-Mar		
		Get one of the pretrained models with sample weights running	1-Mar	12-Mar		
		Determine requirements for retraining with our own modifications and a limited dataset	10-Mar	15-Mar		This will dictate an additional task and timelines for training our own smaller model.
Submit Midterm Report	**22-Mar**	Review past example reports/ask clarifying questions on requirements	2-Mar	12-Mar		
		Complete an outline with bullet points from research so far.	7-Mar	12-Mar		
		Delegate portions of report	12-Mar	14-Mar		
		Finalize content and review	15-Mar	20-Mar		
Test ShapeTalk model on 3D Scan data	**12-Apr**	Determine if there are certain shape types we need to collect scanned data for	15-Mar	19-Mar		
		Reserve 3D scanner through Makerspace & gather varied scan results of differing qualities. Likely some that we intentionally cause noisy data or more imperfections.	20-Mar	2-Apr		Spring break is Mar 23-Mar 31 (either will be catch up time or time where progress is slow)
		Test pretrained weights with our point clouds from the 3d scanner	2-Apr	6-Apr		
		Assuming we did some targeted model retraining - test	4-Apr	10-Apr		
		Draw conclusions from tests	10-Apr	12-Apr		
Present our Project to the class	**19-Apr**	Review presentation requirements once published	3-Apr	6-Apr		
		Start creating outline of Slide Presentation	10-Apr	12-Apr		
		Determine if we will do any live demoing with Jupytery - if so prep demo	10-Apr	16-Apr		
		Complete presentation materials	10-Apr	17-Apr		
Project Website	**05/03**	Determine how to host the website	27-Feb	3-Mar		
		Set up domain	3-Mar	6-Mar		
		Create page skeleton	6-Mar	2-Apr		
		Review published assignment and requirements	3-Apr	6-Apr		
		Add information from midterm report	23-Mar	2-Apr		
		Determine key visuals to best display our results	6-Apr	17-Apr		
		Embed interactive and visual content on site and finalize language on site	17-Apr	1-May		

References

[1] Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., & Vondrick, C. (2023). *Zero-1-to-3: Zero-shot One Image to 3D Object*. arXiv preprint arXiv:2303.11328.

[2] Abdelreheem, A., Skorokhodov, I., Ovsjanikov, M., & Wonka, P. (2023). *SATR: Zero-Shot Semantic Segmentation of 3D Shapes.* Retrieved from arXiv preprint: arXiv:2304.04909 [cs.CV]

[3] Achlioptas, P., Huang, I., Sung, M., Tulyakov, S., & Guibas, L. (2023). *ShapeTalk: A Language Dataset and Framework for 3D Shape Edits and Deformations*. Conference on Computer Vision and Pattern Recognition (CVPR). https://openaccess.thecvf.com/content/CVPR2023/papers/Achlioptas_ShapeTalk_A_Language_Dataset_and_Framework_for_3D_Shape_Edits_CVPR_2023_paper.pdf

[4] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). *Learning Transferable Visual Models From Natural Language Supervision.* Retrieved from arXiv preprint arXiv:2103.00020.

[5] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). *Hierarchical Text-Conditional Image Generation with CLIP Latents.* arXiv preprint arXiv:2204.06125. [cs.CV]

[6] Sanghi, A., Chu, H., Lambourne, J. G., Wang, Y., Cheng, C.-Y., Fumero, M., & Malekshan, K. R. (2022). *CLIP-Forge: Towards Zero-Shot Text-to-Shape Generation.* arXiv preprint arXiv:2110.02624. [cs.CV]

[7] Jain, A., Mildenhall, B., Barron, J. T., Abbeel, P., & Poole, B. (2022). *Zero-Shot Text-Guided Object Generation with Dream Fields.* In CVPR.

[8] Fu, R., Zhan, X., Chen, Y., Ritchie, D., & Sridhar, S. (2023). *ShapeCrafter: A Recursive Text-Conditioned 3D Shape Generation Model.* arXiv preprint arXiv:2207.09446. [cs.CV]

Contributions:

- **Abhinav**: Background, Method (Major contributor) and review of Introduction, Experiments (Minor contributor)
- **Kevin:** Dataset (Major contributor), Timetable and Method (Minor contributor)
- **Laura:** Motivation, Timetable, References (Major contributor), review of background review of experiment. (Minor contributor)
- **Keshav:** Evaluation, Experiment (Major contributor) and review of Introduction. (Minor contributor)