Multimodal 3D Shape Retrieval with Sketch and Text

Xiang Li and **Yuyao Huang** and **Ethan Chau** and **Zaid Shahid** and **Ashwin Kumar**
EECS 486 Final Project, University of Michigan
{xkevli, yuyaoh, echau, zshahid, ashwinak}@umich.edu

1 Project Description

Recent progress of virtual reality (VR) technologies, like the Meta Quest 3, is pushing businesses to establish their presence in the Metaverse, a realm where 3D models serve as essential building blocks. In this digital space, accurately matching a company's branding and thematic elements with suitable 3D models is crucial yet challenging, primarily due to the limitations of current search technologies. Most 3D modeling platforms rely on text-based searches, where descriptions can be ambiguous, sparse, or misaligned with the visual features of the models, leading to prolonged and inefficient retrieval processes.

Recent advancements in neural networks, particularly in sketch-based 3D shape retrieval as explored by (Gao et al., 2020), offer promising new directions. However, current evaluation datasets such as SHREC'14 (Li et al., 2014) are insufficient for today's detailed retrieval demands due to their broad and often imprecise categorizations of sketches and 3D shapes. For instance, one of our team members experienced significant delays—over 20 minutes and nearly a hundred searches—to locate a specific model of a recycle bin on SketchFab, a popular 3D shape search engine. This incident underscores the critical need for improved search methodologies.

To address these challenges, we are developing a novel evaluation dataset based on the ShapeNetCore repository. This dataset includes detailed shape descriptions and simulated user queries generated by two distinct Large Language Models (LLMs), accompanied by sketches derived from an advanced image-to-sketch neural model. We also provide manually-written captions and hand-drawn sketches for comparison. Our goal is to capture the visual and structural essence of 3D shapes more accurately than traditional text descriptions, facilitating a more efficient and precise retrieval process.

Moreover, we propose a new approach to utilize off-the-shelf multimodal embeddings for 3D shape retrieval, by generating 2D renderings of 3D models from multiple perspectives, selecting the most relevant snapshots through a multimodal search algorithm, and correlating these images back to their original 3D shapes. Such a strategy leverages both visual and textual data, significantly enhancing the accuracy of 3D shape searches. It also allows us to use existing models without the need for any training or fine-tuning.

Our project not only aims to refine the academic exploration of multimodal 3D model retrieval but also seeks to offer a practical solution that caters to the operational needs across various industries:

- **Gaming and Virtual Reality**: Game developers and VR content creators can significantly shorten their asset discovery and integration process, allowing for faster development cycles and more detailed, visually consistent worlds.
- **Architecture and Engineering**: Professionals in these fields can quickly locate and utilize 3D models for simulations, visualizations, and presentations, improving project workflows and client engagements.
- **E-commerce and Retail**: Online retailers can employ advanced 3D model search capabilities to offer customers more accurate visualizations of products, enhancing user experience and engagement.
- **Education and Training**: Educational content developers can find appropriate 3D models to create immersive and interactive learning environments for subjects such as science, history, and art.

By creating a richer, more representative evaluation dataset and a robust retrieval framework, we intend to create a practical retrieval solution for 3D creators.

2 Related Works

The field of 3D shape retrieval has seen substantial advancements through deep learning technologies, which have progressively enhanced model accuracy and search efficiency. We review key developments that have set the stage for our research, identifying areas where our work can extend existing knowledge.

ModelNet and Multi-View CNNs: The introduction of the ModelNet dataset by Wu et al. (Wu et al., 2015), accompanied by the use of a convolutional deep belief network, marked a significant advancement in 3D shape retrieval, achieving a 49.94% AUC. The subsequent improvement by Su et al. (Su et al., 2015), through the implementation of a Multi-View CNN that assimilates multiple perspectives of a 3D shape, escalated retrieval accuracy to 80.2%. This approach highlighted the untapped potential of leveraging extensive 2D image data for pretraining CNNs, as opposed to relying solely on more limited 3D shape data. Inspired by this, our project utilizes similar principles to enhance the multimodal integration of text and sketches for 3D model retrieval, building on the foundation that 2D representations can significantly aid the interpretation of 3D structures.

Interactive Attention Modules: Gao et al. (Gao et al., 2023) further refined 3D shape retrieval by integrating CNNs with interactive attention modules to extract detailed semantic features. To avoid the cost of training a new model for retrieval, we apply off-the-shelf commercial embedding models and directly benefiting from pretrained multimodal embeddings. Yang et al. also learns a 2D prediction model using 2D inputs generated from 3D models like our paper and others', but does so by combining 2D perspectives before inference (Yang et al., 2022). Our paper keeps 2D sketches separate to account for users drawing sketches from some angles more than others, resulting in higher performance for commonly drawn user perspectives.

New Dataset with Paired Sketches and 3D Shapes: The ModelNet40 dataset, commonly used in prior studies, often showcases higher retrieval performances due to its simplicity and limited category range. This insight has led to critiques about its real-world applicability (Qi et al., 2021). Qi et al. addressed this by introducing a new dataset that pairs sketches with 3D shapes, which however, underscored the high costs and labor-intensive nature of generating quality sketches manually. This challenge directly informs our experimentation with synthetic sketch generation techniques, aiming to reduce costs and scale the creation of high-quality evaluation data for retrieval systems.

Innovative Captioning Techniques: Luo et al. (Luo et al., 2023) proposed a novel approach for 3D model captioning using advanced LLMs like BLIP2 and CLIP for generating captions from multiple views of 3D models, processed by GPT-4 to synthesize comprehensive captions. This strategy showcases the potential of using sophisticated LLMs to enhance the textual description of 3D models. Encouraged by their results, our project also explores the integration of LLM-generated texts but simplifies the process. We would like to explore Luo et al's technique in a future work.

Each of these studies contributes to the scaffold upon which our project is built. By synthesizing their innovations—such as using 2D representations and integrating sophisticated captioning techniques—we aim to develop a more accurate multimodal 3D model retrieval system. Our work seeks not only to explore the academic boundaries but also to provide practical solutions that can be readily applied in industry contexts, thereby filling the gap between theoretical research and real-world application.

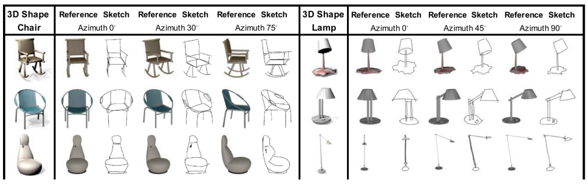
3D Shape Chair
Reference Sketch Azimuth 0°
Reference Sketch Azimuth 30°
Reference Sketch Azimuth 75°
3D Shape Lamp
Reference Sketch Azimuth 0°
Reference Sketch Azimuth 45°
Reference Sketch Azimuth 90°

Figure 1: Samples from Qi et al's dataset

3 Dataset

Existing research datasets for 3D shape retrieval, such as SHREC'14, have been instrumental in advancing the field. However, their limited number of categories and lack of intra-category variations may not accurately reflect the real-world needs of 3D creators. To address this issue, we propose the creation of a new dataset that focuses on a more nuanced retrieval task, aiming to bridge the gap between academic research and practical applications.

We build an evaluation dataset by annotating the camera category of the ShapeNetCore (Chang et al., 2015) dataset, which contains 113 camera models with diverse styles, such as DSLRs, ball-shaped webcams, and CCTV cameras. Four sample camera objects are shown in Figure 2. We choose the camera category because cameras within a subcategory, like DSLR, often share similar visual characteristics, yet possess subtle design differences that are challenging to capture through textual descriptions alone. This mirrors the real-world difficulty faced by 3D creators when searching for assets in a specific style, as the nuances that distinguish one model from another may not be easily conveyed through text-based search queries.

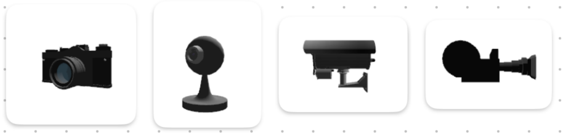

Figure 2: A sample of camera 3D shapes in ShapeNetCore. From left to right: DSLR, ball-shaped webcam, CCTV, classic film camera

For each camera shape, we generate 3 snapshots from angles of 0, 30, and 75 degrees, using the same approach as Qi et al. Then, we generate system descriptions of the snapshots with GPT-4 Vision and simulate user queries with Gemini 1.0 Pro Vision. Manual queries are also written for each of the 113 shapes for comparison. See Figures 3 and 4 for example captions.

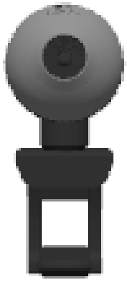

Figure 3: Example of Relevant LLM Captions
Gemini: A grey and black webcam with a round camera lens and a rectangular base.
GPT-4: Black spherical webcam with stand.
Human: Gray spherical webcam with clamp mount.

Additionally, we include sketches of snapshots produced by an image-to-sketch neural model. To simulate varying angles from which a user might draw a sketch, we randomly sample 1 angle out of the 3 to be converted into a sketch for each 3D shape. Manual sketches of the 113 shapes are drawn by hand for comparison (see Figure 5).

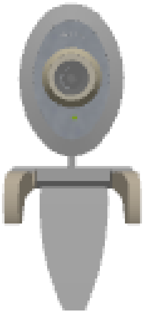

Figure 4: Example of Irrelevant LLM Captions
Gemini: A grey and white cartoon robot with a camera lens for a head.
GPT-4: Gray spaceship cartoon with oval body and three legs.
Human: Tan oval webcam on a mount with two legs and a backplaten.

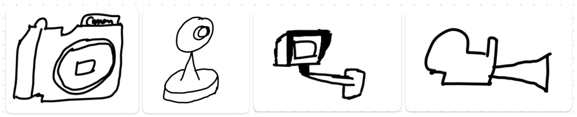

Figure 5: A sample of hand-drawn sketches. From left to right: DSLR, ball-shaped webcam, CCTV, classic film camera

4 Image-to-Sketch Conversion

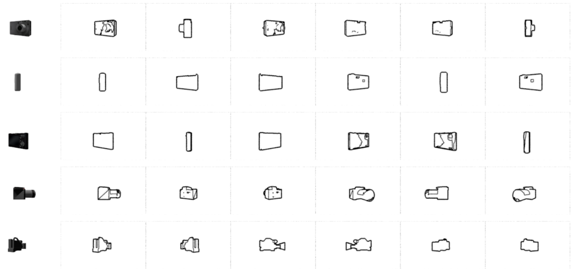

Figure 6: Sketch Generation from Camera 3D Shapes

To transform snapshots of 3D shapes into sketches, we start by pre-processing the input images, converting them into grayscale. This simplifies the images into intensity values indicative of various brightness levels, preparing them for edge detection. Should the dimensions of the grayscale image differ from the original, we resize the color image to ensure a perfect alignment between the final sketch and the original image upon comparison. We conduct edge detection in two stages for enhanced results. In the initial stage, we apply a binary threshold to the grayscale image to highlight the primary outlines. In the subsequent stage, we adjust the threshold dynamically, based on the image's average grayness, to capture finer details in areas where the contrast is less distinct than in the main outlines. We then merge the 5 detected outlines and details into a single image to craft the sketch. A few examples are shown in Figure 6 to demonstrate this technique.

However, it is easy to tell that those sketches are more like edge tracings of the original images, rather than hand-drawn sketches. Hence, we experiment with a neural photo-to-sketch model called CLIPasso (Vinker et al., 2022) to create more realistic sketches. We pick CLIPasso because it can generate sketches with varying levels of abstraction, while preserving their key visual features. Unlike conventional sketching methods, CLIPasso does not utilize a sketch dataset for training, rather it is optimized under the guidance of CLIP (Radford et al., 2021). Thus, CLIPasso is not limited to specific categories observed during training, as no category definition was introduced at any stage. This makes CLIPasso robust to various inputs and capable of generating sketches across a wide range of domains, making it suitable for use in our dataset creation process where objects with varying shapes need to be converted to sketches. A few samples of the sketches generated by CLIPasso are presented in Figure 7. The level of detail in the sketch can be directly controlled by specifying the number of strokes during generation. Compared to sketches generated by OpenCV in Figure 6, CLIPasso's sketches are much more human-like, freeform, and realistic.

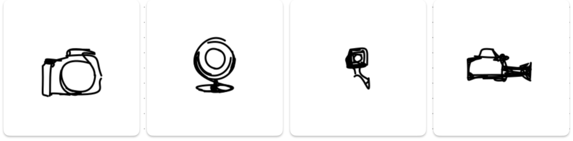

Figure 7: A sample of CLIPasso-generated sketches. From left to right: DSLR, ball-shaped webcam, CCTV, classic film camera

5 Methodology

Building upon the success of the multi-view CNN approach introduced by Su et al. [1], our retrieval system generates 2D snapshots of 3D shapes from various angles. Given a user's text query and sketch query, we rank the models in the dataset by relevance using the snapshots and text descriptions. Due to the presence of 2 modalities, there are several ways to assess the relevance between the user query and the 3D shapes. In this work, we explore 3 different approaches to calculate relevance.

1. **Weighted Sum of Textual and Image Similarity**: We consider the simplest way of combining the text and image modalities by using a weighted sum of the textual similarity between the user text query and the shape description, and the image similarity between the user sketch query and the shape snapshots. We utilize the multimodal embeddings from Google (Google, 2024) and Microsoft (Microsoft, 2024) for the embedding similarity calculations.

2. **Crossmodal Similarity**: We once again use the multimodal embeddings from Google and Microsoft. These embeddings map images and texts into a common latent space, allowing us to directly compute cosine similarity between pairs of user and shape data across modalities: user text with shape snapshot and user sketch with shape description. The similarity between the query and shape is determined by taking the maximum of the two cosine similarities.

3. **Image-to-Text Conversion**: We experiment with converting all modalities into text and compute similarity based only on text. GPT-4 Vision is used to create captions for the user sketches and Gemini 1.0 Pro Vision is used to create captions for the shape snapshots. A single text embedding model is then used to measure the cosine similarity between user sketch description and shape description.

6 Evaluation

Funkhouser et al. evaluate their multimodal 3D shape search engine using median rank and the percentage of target shapes appearing in the top 16 search results (Funkhouser et al., 2003). They categorize searches into 3 types: text-only, sketch-only, and a combination of both. Analyzing these categories separately reveals the effectiveness of combining text and sketch search results into the final output.

In our experiment, we measure the percentage of queries that identifies a target shape in the top k search results, with varying k from 1, 5, to 10. We have a total of 113 queries, one for each 3D shape in the dataset. From now on, we abbreviate the name of this measurement to be percent top k hits. We also vary the weighting for images and texts in the weighted sum approach to investigate the effect of combining text and sketch in 3D shape retrieval. We plot the percent hits for the 3 approaches on the same graph, each approach also has two variants: human or machine annotated. The x axis of the plot shows the image weight, which only applies to the weighted sum approach. The text weight is not shown because it can be easily deduced by subtracting the image weight from 1.0.

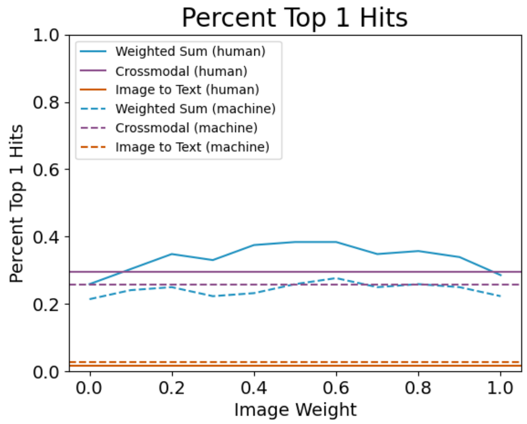

Figure 8: Percentage of queries that identifies a target shape in the top 1 search results using Microsoft Azure embeddings

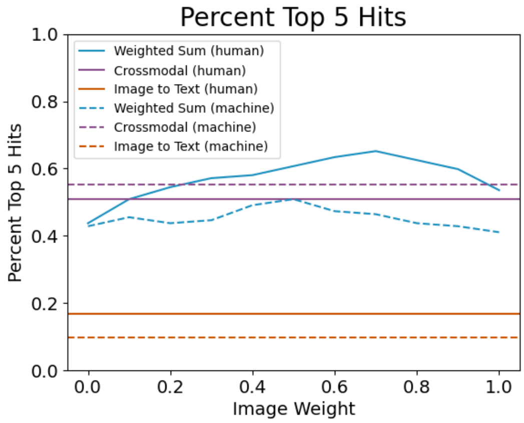

Figure 9: Percentage of queries that identifies a target shape in the top 5 search results using Microsoft Azure embeddings

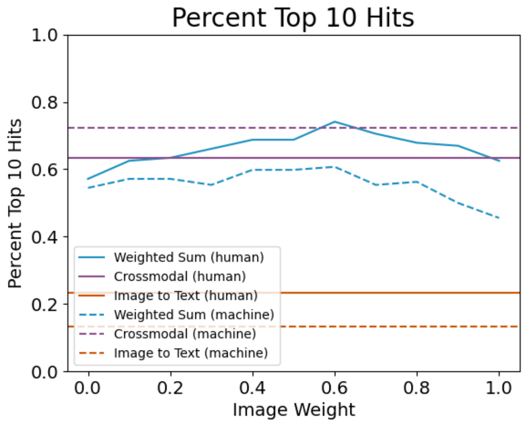

Figure 10: Percentage of queries that identifies a target shape in the top 10 search results using Microsoft Azure embeddings

We identify 3 consistent trends in the experiment results, as shown in Figure 8, 9, and 10:

1. **Weighted sum achieves the highest percent hits on all k values.** We observe that weighted sum achieves the best performance when the image weight is around 0.6, suggesting that the sketches contribute more significantly to search precision than text. Examining the two extremes where the image weight is either 0 (text-only) or 1 (sketch-only), we notice that sketch-only retrieval outperforms text-only retrieval. This discovery is opposite from Funkhouser et al's finding, which shows that text-only retrieval significantly outperforms sketch-only in 3 out of 5 object categories. We think the reason is because modern neural embeddings are much better at extracting useful representations out of images than Funkhouser et al's contour shape descriptors based on manual feature engineering.

2. **Human annotations outperforms machine annotations in all but 3 instances.** We notice that human annotations outperforms machine annotations in all 3 approaches, except in 2 crossmodal cases and 1 image to text case. The underlying reason is that human sketches are generally higher quality than CLIPasso sketches and human queries are also generally higher-quality than GPT-4

Vision queries. In other words, machine annotations underestimate the performance of our retrieval system.

3. **Crossmodal achieves slightly lower percent hits than weighted sum and image to text is much worse than both.** For all human annotated cases, crossmodal consistently lags behind weighted sum. However, crossmodal outperforms weighted sum in 2 machine annotated cases. This again highlights that the sketches and queries generated by models are not guaranteed to share the same distribution as the ones produced by humans. On the other hand, image to text is consistently the worst approach because GPT-4 Vision sometimes have trouble identifying the object in the sketch and produce irrelevant captions. We hypothesize that GPT-4 Vision is primarily trained on photographs and paintings, so sketch captioning presents a challenging out-of-distribution problem for the model.

We also conduct the same experiments for the Google Vertex embedding model. We find similar trends but the average performance to be lower than the Azure model. See Figures 14, 15, and 16 in the appendix for experiment plots.

To investigate why the weighted sum is doing well, we visually cluster all 113 embeddings of the snapshots (Figure 11), hand-drawn sketches (Figure 12), and CLIPasso-generated sketches (Figure 13) using t-distributed Stochastic Neighbor Embedding (t-SNE). We find that cameras of similar shapes are usually clustered close to each other on the plots, showing the effectiveness of off-the-shelf embedding models in extracting useful representations for images and sketches.

Overall, our retreival system performs well. The top weighted-sum setup finds the target shape in the top 1 result about 40% of the time, within the top 5 results about 60% of the time, and within top 10 results about 80% of the time. Utilizing both the image and text modalities in the weighted-sum setup also significantly boosts the percent hits for all k values, as much as 48% when compared to text-only at k=1. See Table 1 for details.

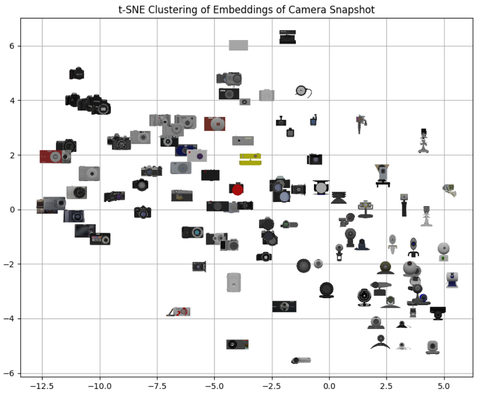

Figure 11: t-SNE Clustering of Embeddings of Camera Snapshots

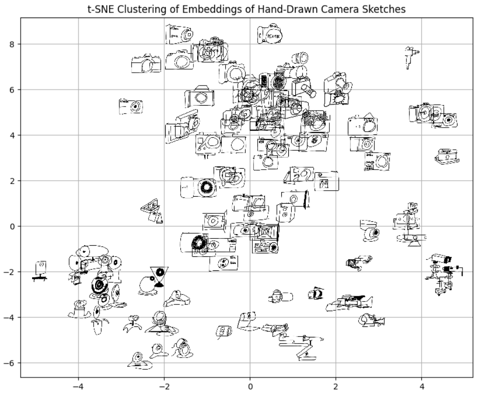

Figure 12: t-SNE Clustering of Embeddings of Hand-Drawn Camera Sketches

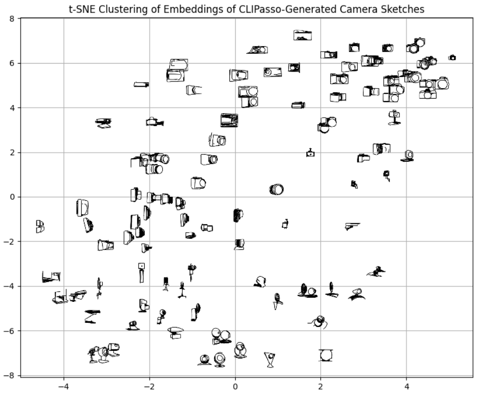

Figure 13: t-SNE Clustering of Embeddings of CLIPasso-Generated Camera Sketches

K	text-only	sketch-only
1	48%	34%
5	39%	13%
10	20%	10%

Table 1: Percent increase in percent hits for different values of k. Text-only and Sketch-only are compared to the weighted-sum approach with an image weight of 0.6 and a text weight of 0.4.

7 Efficiency and Scalability

Our retrieval system is relatively efficient during search time because we only need to calculate embeddings for the user's sketch and text and then do light-weight cosine similarity with precomputed embeddings. However, the indexing process can be costly because we need to use a vision LLM to generate captions for each snapshot. For a dataset with N shapes, we would need to make 3N calls to an LLM api to generate captions for snapshots for each of the 3 angles. Depending on the number of N, this can be prohibitively costly. If some performance sacrifice can be tolerated, LLMs can be replaced with cheaper models like BLIP. Additionally, we also need to generate embeddings for all snapshots, sketches, and captions. However, these embeddings are relatively cheap to produce and only need to be generated once during the indexing phase. New embeddings can be incrementally added to the index as the dataset grows.

8 Conclusion

In this paper, we have presented a multimodal 3D shape retrieval system that integrates both text and sketch queries. By leveraging the ShapeNet-Core dataset and annotating it with system descriptions, user queries, and sketches, we have created a new evaluation dataset that better reflects the real-world needs of 3D creators. Our experiments demonstrate that combining text and sketch modalities significantly improves retrieval performance compared to using either modality alone.

We have explored 3 different approaches to combine text and sketch modalities: weighted sum, crossmodal similarity, and image-to-text conversion. Our results show that the weighted sum approach consistently achieves the highest percentage of top-k hits, with the optimal performance obtained when the image weight is around 0.6. This suggests that sketches contribute more significantly to search precision than text in our retrieval system.

Furthermore, we have found that human annotations generally outperform machine annotations generated by CLIPasso and GPT-4 Vision. Hand-drawn sketches are still needed for accurate performance evaluation. Additionally, we have observed that the crossmodal approach performs slightly worse than the weighted sum approach, while the image-to-text approach is the least effective due to the challenges faced by GPT-4 Vision in captioning sketches accurately.

Our retrieval system demonstrates promising results, with the top weighted-sum setup finding the target shape within the top 1, 5, and 10 picks approximately 40%, 60%, and 80% of the time, respectively. The combination of text and sketch modalities in the weighted-sum approach significantly boosts the retrieval performance compared to using either modality alone.

While our system is efficient during search time, the indexing process can be costly due to the need for LLM-generated captions for each snapshot. Future work could explore more cost-effective alternatives to LLMs, such as BLIP, to improve the scalability of the system.

In conclusion, our multimodal 3D shape retrieval system showcases the potential of combining text and sketch modalities to enhance the accuracy and efficiency of 3D shape retrieval. By creating a new evaluation dataset and exploring various approaches to combine modalities, we have taken a step towards bridging the gap between academic research and practical applications in this field. We hope that our work will inspire future research to further improve the performance and scalability of multimodal 3D shape retrieval systems, ultimately benefiting 3D creators that rely on convenient access to 3D shapes.

9 Work Distribution

1. **Xiang Li**: Responsible for organizing and scheduling team meetings, drafting most of the posters, checkpoints and the final paper, designing and running the 3 approaches in the main experiments, and distributing work between teammates. I'm also responsible for running the CLIPasso model to generate sketches.

2. **Yuyao Huang**: Contributed to the drafting of our research proposal, providing thorough

editing and proofreading for each submission checkpoint to ensure clarity and coherence. I explored several sketch generation programs using neural networks. I developed a Python script utilizing OpenCV to create line drawings from 2D views of 3D models, which was a crucial step in our exploration of neural network capabilities. Additionally, I explored the potential of Microsoft Azure for generating text and image embeddings. This involved not only generating the embeddings but also visualizing them through detailed graphs, which helped us better understand and present our data.

3. **Ethan Chau**: Responsible for looking for ways to generate sketches from images and generating embeddings using Google's Vertex AI. For generating sketches, I mainly used OpenCV to find ways to modify a snapshot of the 3D models into a distinct sketch. I used Google's Vertex AI API to generate text, image, and multimodal embeddings of snapshots, sketches, and captions and evaluated the results by measuring the distance within and between different clusters.

4. **Zaid Shahid**: Responsible for the ImageTo-Text Conversion component, managing everything from querying the Chat GPT and Gemini APIs to generate captions to experimenting with other models like HuggingFace for improved results. I wrote the scripts that facilitated these tasks, playing a key role in enhancing our dataset and the accuracy of our retrieval system. Additionally, I contributed to writing and editing our project's proposal and the final paper, helping to clearly present our research and findings.

5. **Ashwin Kumar**: Responsible for the manual dataset generation, creating human-drawn sketches and captions from the 3 provided perspectives of each of the 113 camera 3D models and suitably organizing them. I sought to make sketches as realistic as possible with respect to someone drawing a 3D model from memory by drawing a variety of perspectives and following a procedure to minimize the difference between the way I drew each sketch. I was also responsible for preliminary research into text-image embedding methods with GPT4V, and I contributed to writing and editing our final project and deliverables.

References

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

Thomas Funkhouser, Patrick Min, Michael Kazhdan, Joyce Chen, Alex Halderman, David Dobkin, and David Jacobs. 2003. A search engine for 3d models. *ACM Transactions on Graphics*, 22(1):83–105.

Kai Gao, Jie Yang, Hongbo Guo, Lin Gao, Biao Li, and Yadong Zheng. 2020. Novel sketch-based 3d model retrieval via cross-domain feature clustering and matching. In Igor Farkaš, Paolo Masulli, and Stefan Wermter, editors, *Artificial Neural Networks and Machine Learning – ICANN 2020*, pages 299–311. Springer International Publishing, Cham.

Xueyao Gao, Wentao Jia, and Chunxiang Zhang. 2023. 3d model retrieval based on interactive attention cnn and multiple features. *PeerJ Computer Science*, 9.

Google. 2024. Get multimodal embeddings | generative ai on vertex ai. `https://cloud.google.com/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings`. Accessed 17 Mar. 2024.

Bo Li, Yijuan Lu, Chunyuan Li, Afzal Godil, Tobias Schreck, Masaki Aono, Qiang Chen, Nihad Karim Chowdhury, Bin Fang, Takahiko Furuya, et al. 2014. Shrec'14 track: Large scale comprehensive 3d shape retrieval. In *Eurographics Workshop on 3D Object Retrieval*, volume 2, page 14. .

Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. 2023. Scalable 3d captioning with pretrained models.

Microsoft. 2024. Multimodal embeddings concepts - image analysis 4.0 - azure ai services. `https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/concept-image-retrieval`.

Anran Qi, Yulia Gryaditskaya, Jifei Song, Yongxin Yang, Yonggang Qi, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. 2021. Toward fine-grained sketch-based 3d shape retrieval. *IEEE Transactions on Image Processing*, 30:8595–8606.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. 2015. Multi-view convolutional neural networks for 3d shape recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 945–953.

Yael Vinker, Ehsan Pajouheshgar, Jessica Y. Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. 2022. Clipasso: Semantically-aware object sketching.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920.

Hairui Yang, Yu Tian, Caifei Yang, Zhihui Wang, Lei Wang, and Haojie Li. 2022. Sequential learning for sketch-based 3d model retrieval. *Multimedia Syst.*, 28(3):761–778.

A Appendix

Percent hits under varying top k using the Google Vertex embeddings.

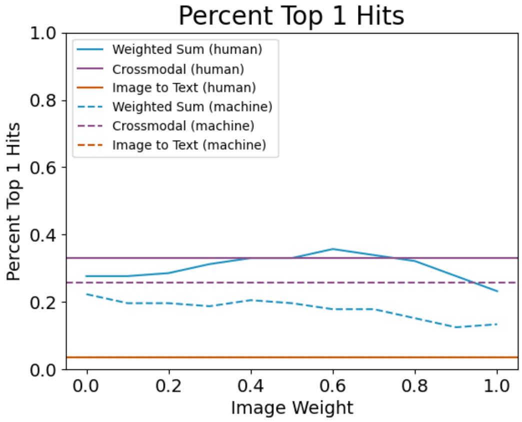

Figure 14: Percentage of queries that identifies a target shape in the top 1 search results using Google Vertex embeddings

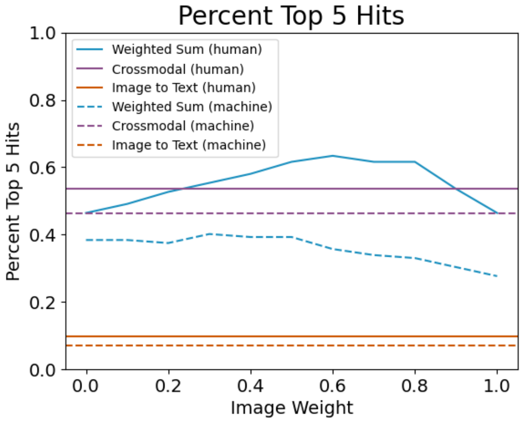

Figure 15: Percentage of queries that identifies a target shape in the top 5 search results using Google Vertex embeddings

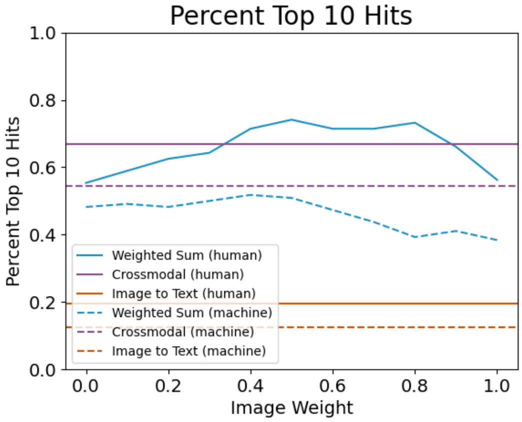

Figure 16: Percentage of queries that identifies a target shape in the top 10 search results using Google Vertex embeddings