Multimodal Concept Mapping

Siddharth Boppana, Tian Yun, Carina Curto, Ellie Pavlick
Brown University

Abstract

Multimodal Large Language Models (MLLMs) have recently seen success in multimodal tasks by harnessing the reasoning capabilities of pre-trained modality-specific encoders and large language models (LLMs). However, MLLMs often fail to generalize to out-of-distribution data, leading to hallucination and over-reliance on knowledge from pre-training. Why does this failure to generalize occur after modality alignment with multimodal fine-tuning? We investigate how MLLMs learn novel concepts by introducing concepts in unimodal pre-training or multimodal fine-tuning, and evaluate the model's ability to generalize between the two settings. By framing novel concepts within an existing concept hierarchy learned from pre-training, we evaluate how *deeply* a model maps concepts across modalities. We find that: (1) MLLMs fail to generalize to novel concepts introduced in multimodal fine-tuning in captioning tasks; (2) The ratio of novel concepts and whether the vision encoder is frozen affect generalization ability; (3) The correlation between the geometry of the model's latent space and a model's ability to generalize improves when the vision encoder is trainable.

1 Introduction

Multimodal models utilize data from different modalities to learn joint representations. Most often, these models focus on learning a mapping between vision and language to solve a diverse set of tasks including captioning, visual question answering (VQA), and semantic segmentation. Recent developments in multimodal learning have led to the widespread adoption of multimodal large language models (MLLMs). (Merullo et al., 2023; Liu et al., 2023; Li et al., 2023; Beyer et al., 2024)

By learning a mapping between a modality-specific encoder and a large language model (LLM), these models can auto-regressively output text based on both visual and textual inputs. However, MLLMs still exhibit the same failure modes of their pre-trained LLMs, while also struggling to map relevant information across modalities. Previous work attempts to either mitigate these issues by improving the vision encoder or by modifying intermediate representations in the shared latent space. Although these methods improve downstream performance by better grounding in visual information, they do not directly evaluate performance on out-of-distribution (OOD) data.

To understand the failure mode of MLLMs when generalizing to OOD data, we investigate how MLLMs learn novel concepts during multimodal fine-tuning. If the vision encoder and LLM have never been exposed to a certain concept in pre-training, can they learn a representation of this concept during fine-tuning which fits into their existing concept hierarchy? We answer this question by systematically withholding both individual concepts during unimodal pre-training, and introducing them during fine-tuning.

Our approach is conducted in the two stages of training MLLMs: unimodal pre-training and multimodal fine-tuning. First, we pre-train the vision encoder from scratch on a subset (X) of a dataset. During pre-training, we exclude images corresponding to certain concepts (Y). Then, we fine-tune the MLLM with the entire dataset (X + Y). Once fine-tuned, we evaluate captioning and VQA accuracy on images with concepts introduced in pre-training vs. fine-tuning. To ensure that concepts remain novel to the LLM as well, we replace the token of a concept with an out-of-distribution token. We introduce this approach in a toy setting with synthetic images of shapes, where we withhold primary visual concepts like specific colors, shapes, and sizes. We then evaluate whether the MLLM can associate concepts within a shared hierarchy when prompted, even if the hierarchical information is not explicitly provided. We use dimensionality reduction techniques to further investigate

the shared vision-language latent space, to determine if there is a geometric structure corresponding to how we understand a concept hierarchy.

We show that MLLMs fail to generalize as well to concepts introduced in fine-tuning than pre-training, achieving worse performance on both existing learned concepts and fully out-of-distribution concepts within a shared concept hierarchy. We then evaluate how two training decisions from previous vision-language literature affect the depth of a concept mapping: which components of the MLLM are frozen during fine-tuning and ratio of novel concepts introduced during fine-tuning. We find that both of these training decisions affect the concept mappings between novel and previously seen concepts. Finally, we investigate whether the geometry of the MLLM's latent space can be used to understand concept mappings, by quantifying how concepts cluster within the latent space at various layers. We find that a MLLM's ability to generalize does not correlate to where concepts lie in its latent space, meaning that models may be representing hierarchical relationships in other ways.

2 Related Work

2.1 Multimodal Large Language Models

Multimodal learning architectures have recently converged to the multimodal large language model, focusing specifically on vision-language models. A MLLM contains three parts: a pre-trained vision encoder, a pre-trained LLM, and an adaptor network. Hidden representations from the vision encoder are passed through the adaptor into the LLM's embedding space, allowing for the input to contain both language and vision. The adaptor network is typically a one-layer linear projection or MLP, but can also consist of several MLPs or even a transformer model. Training MLLMs requires paired image-text data with an autoregressive objective like the base LLM, where each patch of the original image can be thought of as a vision token. Previous work on improving MLLMs has focused on the vision encoder as a bottleneck for solving multimodal tasks. CLIP-based vision encoders have been used to allow for language supervision during pre-training as well (Liu et al., 2023), while other work uses self-supervised methods to better learn spatial information. (Merullo et al., 2023; Fan et al., 2025) These models are first multimodally fine-tuned on a variety of vision-language tasks including captioning, VQA, segmentation, and counting. Another training decision is to freeze the vision encoder, both to reduce training time and to preserve the visual representation from pre-training. (Beyer et al., 2024; Zhai et al., 2022) However, it is unclear how this affects intermediate representations and whether this ultimately inhibits learning.

2.2 Multimodal Interpretability

Previous work has also shown that the underlying representations of multimodal models are interpretable, using methods similar to those of existing transformer-based models. With the rise of dictionary learning as an interpretability technique in LLMs (Bricken et al., 2023), features are used as the primary building block to understand model behavior. (Luo et al., 2024) show that task vectors used in LLMs can be used for MLLMs as well to solve in context learning tasks with both images and text by steering on a relevant feature. Others also develop frameworks to find cross-modal features explaining model behavior on maximally activating samples. (Gandelsman et al., 2025; Parekh et al., 2024), MLLMs have been found to be blind to out-of-distribution shapes and other spatial relations, which is largely attributed to their inability to reason past basic relations. (Rudman et al., 2025; Rahmanzadehgervi et al., 2024) Modifying the intermediate representations in vision-language models can improve grounding, using LLM interpretability methods like steering vectors. (Jiang et al., 2025)

3 Methods

3.1 MLLM Architecture

We adopt a similar architecture to (Liu et al., 2023), and use a pre-trained vision encoder and LLM with a learned adaptor between the two. For the vision encoder, we use the ViT-Base model (Dosovitskiy et al., 2021) and train from scratch. For the LLM, we use a pre-trained model from the OPT series (OPT-125M) (Zhang et al., 2022). We choose a single MLP as the adaptor, as it is the method most common in the literature while maintaining simplicity, which allows for successful transfer of visual information. When varying frozen parameters, we experiment in two settings during multimodal fine-tuning: freezing both the vision encoder and LLM, and freezing only the LLM .

3.2 Datasets

Synthetic Shapes

Visual Concept	Types
Shape	Circle, Triangle, Square, Diamond, Star
Color	Red, Orange, Yellow, Blue, Green, Purple, Black
Size	Small, Medium, Large

Table 1: Visual concepts used to create shape images and the types per concept included.

We introduce a toy dataset of synthetic shapes to control for visual concepts learned by the vision encoder. We define each class as a combination of three visual properties: shape, size, and color. We synthetically generate 600 images for each of the 105 classes, providing a total of 63000 images in the dataset. The shape and color are discrete properties which don't vary within their class, while each size varies within a range. Shapes are randomly positioned and rotated to encourage a robust visual classifier. 100 images per class are saved as the test set (1050 images), while the remaining are used during pre-training and fine-tuning. A 80/20 split is used between pre-training and fine-tuning. The pre-training dataset excludes certain classes of shapes, while the fine-tuning dataset contains every combination. The full list of visual concepts can be seen in Table 1. By controlling for basic visual concepts, we construct a compositional hierarchy, where two given shapes may share zero to three properties.

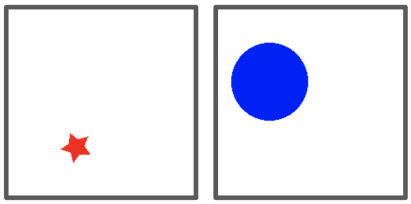

Figure 1: Two examples from our dataset.

3.3 Concept Hierarchies in Latent Space

We define a metric to evaluate whether the latent space of a MLLM encodes novel and previously seen concepts within a larger concept hierarchy geometrically. A MLLM contains a vision encoder M_v and LLM M_l with respective hidden dimensions d_v and d_l, along with a connector which learns a mapping function $f : \mathbb{R}^{d_v} \to \mathbb{R}^{d_l}$. To measure this in a latent space, we compare the distance between embeddings of two different concepts to the similarity of the concepts within a shared concept hierarchy. For a given class C, the embedding of the class in latent space can be computed as the intermediate representation during a forward pass of the MLLM. An image is represented as a series of embeddings due to the nature of the transformer architecture, leading to t vision tokens which are converted to embeddings. This results in t vectors of dimension d_v or d_l, depending on whether the embedding is taken before or after the connector layer. To have a single vector per image, the t vectors are averaged. To determine if concepts' latent spaces align with how similar they are in a pre-determined shared hierarchy, we compute two distance metrics and check if there is a correlation between the two.

For each pair of classes C_1 and C_2 in a given hierarchy H, we define the Hierarchical Distance (HD) and Latent Distance (LD). HD $= \min(n_1, n_2)$, where n_x is the number of shared visual properties between the two types of shapes, which can vary between 0 to 3. LD is defined as the distance between embeddings in space for C_1 and C_2 after a layer L in the MLLM. We choose the Euclidean distance as the distance metric between embeddings, but find similar results with cosine and Manhattan distance as well.

4 Experiments

4.1 Training Details

We train a suite of MLLMs using the architecture outlined in 3.1 and the dataset from 3.2. First, the vision encoder is pre-trained on a subset of the dataset with a classification objective, where each image is mapped to a unique class based on its combined visual properties (e.g., a small red square is treated as a distinct class, not separate size, color, and shape concepts). We train four vision encoders, each with a different amount of data diversity. Of the fifteen primitive visual concepts across the three visual concepts, we pre-train with either zero, two, four, or eight concepts excluded. Each vision encoder is trained for 30 epochs with a learning rate of 1e-4, and the checkpoint with the best validation accuracy is kept for further training and evaluation.

Next, the MLLM is fine-tuned using a pre-trained vision encoder, the adaptor, and pre-trained LLM. The multimodal fine-tuning process uses paired image-caption data, where each image is paired with text that represents some aspect of its visual properties. Concepts introduced in pre-training are captioned with any of the captions in

Data Diversity	Concepts Excluded
0/15	n/a
2/15	Triangle, Orange
4/15	Triangle, Orange, Purple, Medium
8/15	Triangle, Circle, Orange, Purple, Green, Black, Small, Large

Table 2: Excluded visual properties per data diversity setting.

Fine-tuning Prompt Types
"What is this?": " <size> <color> <shape>"
"What color is this?": " <color>"
"What size is this?": " <size>"
"What shape is this?": " <shape>"
"Is this shape <rand_color>?": " <yes_no>"
"Is this shape a <rand_shape>?": " <yes_no>"
"Is this shape <rand_size>?": " <yes_no>"

Table 3: Visual concepts used to create shape images and the types per concept included. The bottom three captions randomly sample an example property to evaluate if the model can recognize whether an image contains the property or not.

Table 3, while novel concepts introduced in fine-tuning are only captioned with a full description (the top row of Table 3).

For novel concepts introduced in fine-tuning, we swap the original name of the concept with an arbitrary token to ensure that there is no learned semantic to the concept in the LLM. This removes any prior association that the model may have with a concept to ensure it is out-of-distribution. We set each novel concept to a string of three random characters that is represented as a single token. Once fully trained, the MLLM is queried about the concepts of images in the held-out test set. We train one set of MLLMs where both the vision encoder and LLM are frozen, while the other set has an unfrozen vision encoder.

4.2 Evaluating Concept Representations

Once a MLLM is trained, we evaluate whether the geometric representation of its latent space is related to its ability to generalize to novel concepts. We begin by gathering hidden representations of each example within the test set, and averaging across examples of a concept to get a single representation per class. Hidden representations are gathered at two different places: the hidden representation before the final class. To begin our analysis on the geometry of the latent space, we use dimensionality reduction techniques (McInnes et al., 2020) to observe how classes cluster based on their visual concepts. We next compute the average Hierarchical Distance and Latent Distance between every pair of classes, and compare the two in the different training settings.

5 Results

5.1 Training

As can be seen in Figures 2, 3, and 4, all models successfully learn to solve the task. During vision pre-training, encoders trained with more data diversity and size perform with nearly perfect accuracy, while the model with the least data diversity and size performs worse, while still better than chance. During multimodal fine-tuning, the MLLMs are able to predict the corresponding visual properties for the various prompts regardless of training diversity and lack of trainable parameters in the LLM. Generally, allowing the vision encoder to train improves performance as expected but still does not enable the models to solve the task perfectly.

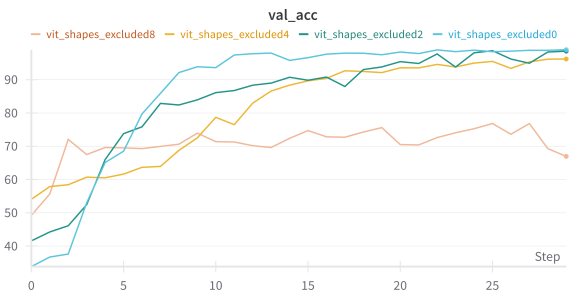

Figure 2: Validation accuracy for the four different ViT runs, each with a different data diversity. As training diversity and dataset size increase, so does validation accuracy.

5.2 Evaluation

Once the models are trained, we evaluate them on the held-out test set, which is split per model on in-distribution (introduced in pre-training) and out-of-distribution (introduced in fine-tuning) examples. We show results on MLLMs in Figure 5. As data diversity introduced in pre-training decreases, accuracy on in-distribution concepts dips slightly, indicating that a mapping is successfully learned from vision to language, and a weak vision

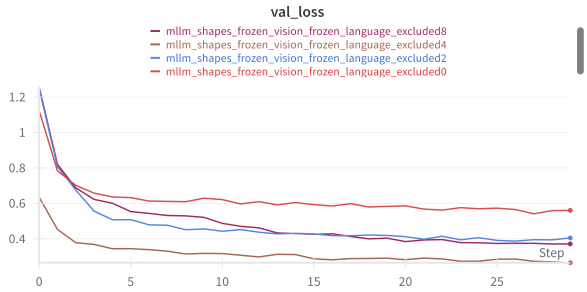

Figure 3: Validation loss for four different MLLM runs, with a frozen vision encoder and frozen LLM. Each run corresponds to a vision encoder with a different pre-training data diversity.

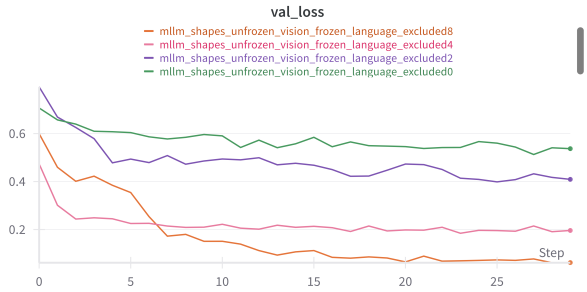

Figure 4: Validation loss for four different MLLM runs, with only a frozen LLM. Each run corresponds to a vision encoder with a different pre-training data diversity.

encoder's accuracy will carry over into the MLLM as well. However, accuracy on out-of-distribution concepts drops sharply despite all four models being fine-tuned on the same amount of data as well. Each model is tested on concepts which were out-of-distribution for the pre-trained vision encoder, meaning that the set of concepts vary across excluded settings. This indicates that data diversity in the fine-tuning stage is not enough to learn generalizable concept mappings. Lack of data diversity in the individual modality encoders harms downstream performance, even when the novel concepts are introduced in fine-tuning.

Another trend arises when the compositionality of the novel concept is accounted for. To analyze this, we further split novel classes into how out-of-distribution they are compared to concepts introduced in pre-training and plot this in Figure 10. As the number of out-of-distribution concepts in a class decreases, so does accuracy in identifying properties. Once again, we see the same trend emerge across data diversity settings, finding that vision models exposed to more concepts in pre-training perform better, even on classes with no

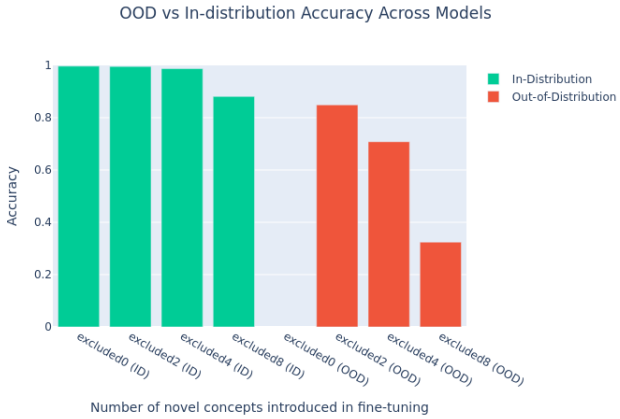

Figure 5: Test set accuracy for in-distribution and out-of-distribution classes. Note that in the excluded0 setting, there are no out-of-distribution concepts, hence the empty bar.

primitive visual concepts from pre-training.

Figure 6: Test set accuracy for classes, split by the number of out-of-distribution properties. Note that in the excluded0 setting, there are no out-of-distribution concepts, hence the empty bars for degrees 1-3. This is also true for degree 3 in the excluded 2 setting.

This phenomenon can be seen further in an example of analyzing accuracy on a single property. Figure 7 shows accuracy for the pre-trained vision encoder vs. the fine-tuned MLLM on the colors introduced in their respective stages of training. The vision encoder solves the task perfectly (defined as correctly choosing a class that contains the correct color), while the MLLM sees a drop-off in performance across all colors. This is an indicator that the MLLM exhibits worse performance on the same concepts due to being exposed to novel combinations of concepts. For example, the model was

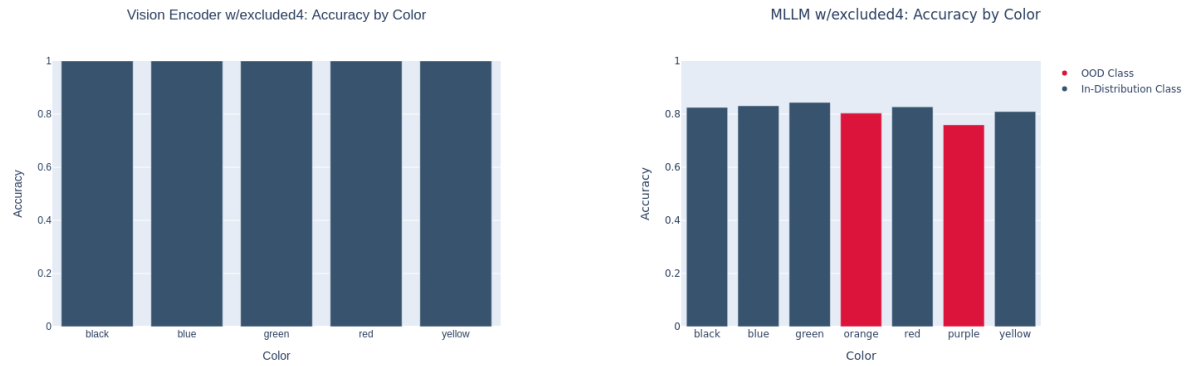

Figure 7: Test-set color accuracy: (left) pre-trained vision encoder, (right) fine-tuned MLLM. Out-of-distribution classes are highlighted in red.

pre-trained with small black squares but not small black triangles. This lack of transfer for the concept black shows that the ability to generalize is not guaranteed to arise during multimodal fine-tuning, even if the fine-tuning dataset includes every combination of property. Not only is the model unable to categorize new concepts, it also struggles to categorize previously seen concepts in new contexts.

5.3 Representation Analysis

Next, we look at the latent space of these models to better understand if its geometry can be interpreted based on its performance and ability to generalize. We begin by computing the embeddings of images from the test set and computing the Latent and Hierarchical distances between all pairs of classes. Figure 8's left plot contains the pairwise Hierarchical distances, while the middle contains the LD plot for embeddings computed after the last layer of the vision encoder kept frozen during fine-tuning. Both plots order classes on the x and y axes alphabetically, resulting in organization by size, color, then shape. We use HD as a ground truth to compare different LD plots to in order to verify whether the latent space matches our intuitive relations between objects. While we observe some structure within the LD plot with a frozen vision encoder, the plot does not correspond with HD aside from certain vertical and horizontal lines corresponding to shapes which share a property.

However, structure emerges when the vision encoder is allowed to train during multimodal fine-tuning. The diagonals on the plot corresponding to classes with a single different property are much closer comparatively than the setting with a frozen vision encoder. There is also a much clearer increase in latent distance as the objects switch properties, as is seen by the blocks corresponding to the same size and color, but various shapes. The simplicity of the HD plot is also a result of treating different sizes, colors, and shapes as categorical variables, when in reality these values are not always separable. Size and color can be interpolated, while certain shapes are more similar to one another. These properties may leave artifacts in the latent distance plots as well.

Two other variations were analyzed when computing the latent distances between classes. First, distances were computed across models with different amounts of data diversity during pre-training. Next, we used embeddings taken directly after the connector, as opposed to before. In both cases, LD plots resembled the middle plot in Figure 8's middle plot with less clear structure. This indicates that while data diversity may benefit generalization, it does not lead to a more interpretable latent space that corresponds to our understanding of concept hierarchies. When experimenting with different distance metrics, we found no noticeable difference in plotting with cosine distance as opposed to Euclidean distance as well.

Finally, we also attempt to manually inspect the latent space by using dimensionality reduction techniques to identify geometric structure. We generally identify clusters corresponding to size, shape, and color across models with different pre-training data diversities, frozen/unfrozen vision encoders, and embeddings from before/after the connector. However, models that generalize better have no discernible differences between their clusters. UMAP plots are also sensitive to initialization, and it is unclear whether the geometric structure can be clearly

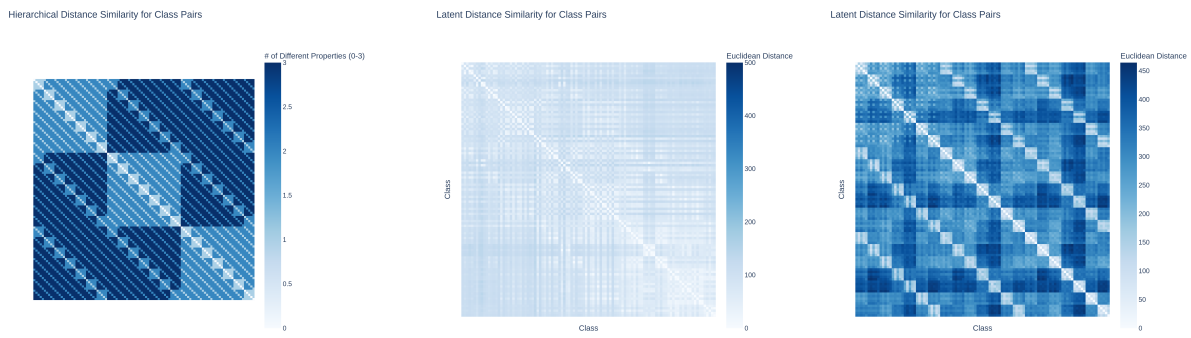

Figure 8: (Left) Hierarchical distance, (Middle) Latent Distance with frozen vision encoder, (Right) Latent Distance with unfrozen vision encoder.

understood in just two dimensions. We include a single set of plots, colored by the visual properties to show some structure. The model used is pre-trained with four concepts excluded, and the embeddings are computed prior to the connector. We see some evidence of pre-trained vision encoders "slotting" in colors (orange and purple) and size (medium) into their embedding space despite not being trained on them.

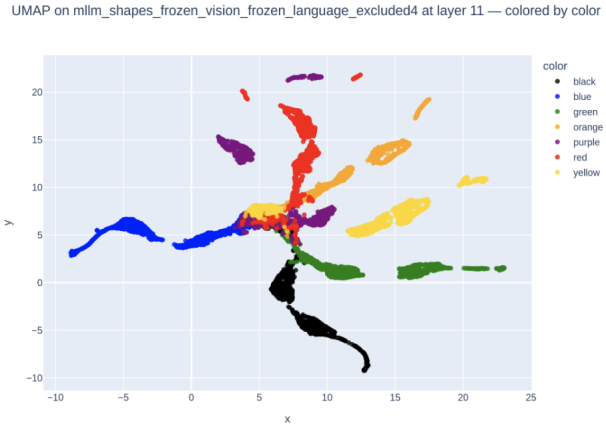

Figure 9: UMAP plot of embeddings computed colored by the original color of the shape.

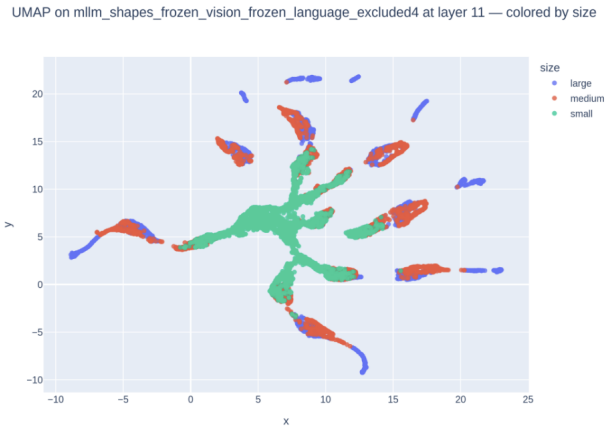

Figure 10: UMAP plot of embeddings computed colored by the original size of the shape.

6 Conclusion

In this work, we show that Multimodal Large Language Models (MLLMs) fail to generalize to novel concepts when their individual modality encoders are not exposed to a wide enough variety of concepts during pre-training. We find that increasing dataset diversity and size leads to a significant improvement in generalizability, both on individual concepts and combinations of concepts. We also find that MLLMs struggle to generalize learned concepts to new contexts, showing that robust vision encoders are still required and cannot be easily corrected during multimodal fine-tuning. Finally, we show that the geometry of a model's latent space does not directly correlate with the existing hierarchy of concepts on our synthetic shapes task. We provide evidence that fine-tuning the visual encoder during multimodal fine-tuning improves alignment between the latent space of the model and existing hierarchy.

References

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. 2024. Paligemma: A versatile 3b vlm for transfer. *Preprint*, arXiv:2407.07726.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. *Preprint*, arXiv:2010.11929.

David Fan, Shengbang Tong, Jiachen Zhu, Koustuv Sinha, Zhuang Liu, Xinlei Chen, Michael Rabbat, Nicolas Ballas, Yann LeCun, Amir Bar, and Saining Xie. 2025. Scaling language-free visual representation learning. *Preprint*, arXiv:2504.01017.

Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. 2025. Interpreting the second-order effects of neurons in clip. *Preprint*, arXiv:2406.04341.

Nick Jiang, Anish Kachinthaya, Suzie Petryk, and Yossi Gandelsman. 2025. Interpreting and editing vision-language representations to mitigate hallucinations. *Preprint*, arXiv:2410.02762.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *Preprint*, arXiv:2301.12597.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Grace Luo, Trevor Darrell, and Amir Bar. 2024. Task vectors are cross-modal. *Preprint*, arXiv:2410.22330.

Leland McInnes, John Healy, and James Melville. 2020. Umap: Uniform manifold approximation and projection for dimension reduction. *Preprint*, arXiv:1802.03426.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. 2023. Linearly mapping from image to text space. *Preprint*, arXiv:2209.15162.

Jayneel Parekh, Pegah Khayatan, Mustafa Shukor, Alasdair Newson, and Matthieu Cord. 2024. A concept-based explainability framework for large multimodal models. *Preprint*, arXiv:2406.08074.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. 2024. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 18–34.

William Rudman, Michal Golovanevsky, Amir Bar, Vedant Palit, Yann LeCun, Carsten Eickhoff, and Ritambhara Singh. 2025. Forgotten polygons: Multimodal large language models are shape-blind. *Preprint*, arXiv:2502.15969.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. 2022. Lit: Zero-shot transfer with locked-image text tuning. *Preprint*, arXiv:2111.07991.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models. *Preprint*, arXiv:2205.01068.