Making CLIP Features Multiview Consistent

Student: Lara Nonino
Supervisors: Barath Daniel, Engelmann Francis
Practical Work - June 2024
Computer Vision and Geometry Group, ETH Zurich, Switzerland

I. Introduction

Advancements in artificial intelligence and computer vision have led to the development of powerful models capable of understanding and interpreting visual and textual information. Among these models, CLIP (Contrastive Language-Image Pretraining) [1] has garnered significant attention due to its ability to connect images with text prompts bidirectionally. This capability allows CLIP to perform various tasks such as zero-shot learning, where the model can understand and generate predictions for previously unseen data. However, despite its effectiveness, CLIP exhibits inconsistencies in its feature representations when dealing with objects viewed from different perspectives. These inconsistencies affect its performance in tasks requiring multiview consistency, such as object recognition and scene understanding. Addressing this limitation is crucial for enhancing the applicability of CLIP in real-world scenarios. This project aims to enhance CLIP's multiview consistency by leveraging SigLIP, a more efficient variant, to improve applications such as object recognition, scene understanding, and augmented reality, where consistent feature representation is paramount.

II. Problem Definition and Literature Review

The primary issue addressed in this project is the lack of multiview consistency in the feature representations generated by CLIP. When presented with images of objects captured from various viewpoints, CLIP fails to produce consistent feature representations for the same objects. This inconsistency poses a significant challenge in applications where accurate recognition and understanding of objects across different perspectives are crucial. Without multiview consistency, CLIP's performance in tasks such as 3D object recognition, scene understanding, and augmented reality applications is compromised. Therefore, there is a critical need to develop methods that enhance CLIP's ability to maintain consistent feature representations across multiple views of the same object.

Challenges associated with this problem include the inherent difficulty in ensuring that features remain stable across different perspectives, the computational complexity of processing multiview data, and the need for large-scale datasets that provide diverse viewpoints of objects. Additionally, achieving multiview consistency without sacrificing the generalization capabilities of the model remains a significant hurdle. Addressing these challenges requires innovative approaches to model training and data processing that can effectively integrate multiview information.

A. Literature Review

Several works have explored the issue of multiview consistency in visual models. A notable example is the research on ConsistNet, which introduces a plug-in module to enhance 3D consistency in diffusion models, allowing them to generate multiview consistent images from a single reference image efficiently [2]. This method shows promise in generating coherent images from multiple perspectives, which is essential for applications like 3D modeling and animation. Similarly, TexPainter addresses the challenge of generating consistent 3D textures by employing a method that integrates sampled views to paint textures. Despite its effectiveness, TexPainter still faces issues such as seams and noise, which can degrade the quality of the final output [3].

In the domain of 3D scene reconstruction, PMVC (Promoting Multi-View Consistency) leverages deep features computed from images to mitigate inconsistencies in 3D reconstructions [4]. This approach employs adaptive sampling strategies to ensure fidelity and outperforms current methods in maintaining consistency across views. These advancements indicate a growing interest in improving multiview consistency, yet they also highlight the complexities involved in integrating multiview data into coherent representations.

Furthermore, recent advancements in probing the 3D awareness of visual foundation models highlight the limitations of models like CLIP in encoding 3D properties accurately. Studies have shown that while models like DINOv2 perform well in capturing fine details, CLIP and similar models often lack depth representation capabilities. This limitation impacts their effectiveness in tasks requiring accurate 3D understanding, such as autonomous driving and robotics, where spatial awareness is crucial [5].

B. Our Approach

Despite significant advancements, there remain opportunities to further enhance multiview consistency in visual models. Existing methods often require substantial computational resources or introduce artifacts that degrade the quality of multiview representations. Additionally, many approaches are tailored to specific tasks or models, lacking a generalized solution applicable across various applications.

In this project, we explored the potential of SigLIP, a variant of CLIP, to address these challenges. SigLIP employs a pairwise Sigmoid loss instead of the standard contrastive learning with softmax normalization. This design allows SigLIP to operate solely on image-text pairs without needing a global view of pairwise similarities, which can improve efficiency. Moreover, SigLIP's loss function is more memory efficient, enabling the training of larger batch sizes without additional resources. These characteristics make SigLIP a promising framework for experimentation, particularly in tasks aimed at achieving multiview consistency.

By fine-tuning SigLIP, we aimed to develop a framework that enhances multiview consistency. This exploration included assessing the feasibility and effectiveness of SigLIP's approach in maintaining consistent feature representations across multiple views of the same object. Through this investigation, we sought to understand how SigLIP could be applied to improve the performance of visual models in applications requiring robust multiview feature consistency [6].

III. Workflow

To analyze the behavior of CLIP and SigLIP image embeddings without any fine-tuning, we needed embeddings of objects viewed from different perspectives. This necessitated the creation of a dataset that was later used to fine-tune the projection head of SigLIP. The workflow followed in this project includes:

- **Dataset Creation**: Constructing a dataset from the ScanNet dataset to obtain multiple views of the same objects.
- **Initial Analysis**: Analyzing the image embeddings produced by CLIP and SigLIP to understand their baseline performance.
- **Fine-tuning SigLIP**: Applying a projection head and triplet loss to enhance the multiview consistency of SigLIP embeddings.
- **Experiments**: Conducting various experiments to evaluate the effectiveness of the fine-tuning process.

IV. Dataset Creation

To achieve the goal of training the image encoder of SigLIP to be multiview consistent, we needed images of the same object seen from different viewpoints. For this purpose, we utilized 350 scenes from the ScanNet dataset [7].

ScanNet provides extensive 3D scans of indoor environments, capturing both 2D and 3D data. More specifically, it contains 2.5 million RGB-D frames across 1513 scans, annotated with 3D camera poses, surface reconstructions, and semantic segmentations. The dataset includes instance-level semantic segmentations and is marked in 20 classes of annotated 3D voxelized objects, providing approximately 90% surface coverage of indoor scenes. This extensive coverage and rich annotation make ScanNet a valuable resource for tasks requiring detailed 3D information and diverse viewpoints.

The primary advantage of using ScanNet is its comprehensive nature, which includes richly annotated data that is indispensable for advanced computer vision tasks. Each scene offers a high level of detail, enabling precise segmentation and accurate projection of objects from multiple viewpoints. By leveraging the annotations and the diverse perspectives provided by ScanNet, we were able to create a robust dataset essential for training the image encoder of SigLIP to achieve multiview consistency. This, in turn, improves the model's performance in tasks requiring consistent feature representation across various viewpoints.

A. Scene Selection and Preparation

We started by selecting 350 scenes from the ScanNet dataset, which offers an extensive array of indoor environments. Each scene in ScanNet includes a detailed 3D scan with segmented meshes for each object present in the scene (see Figure 2). This segmentation is crucial for isolating individual objects and projecting them into different viewpoints.

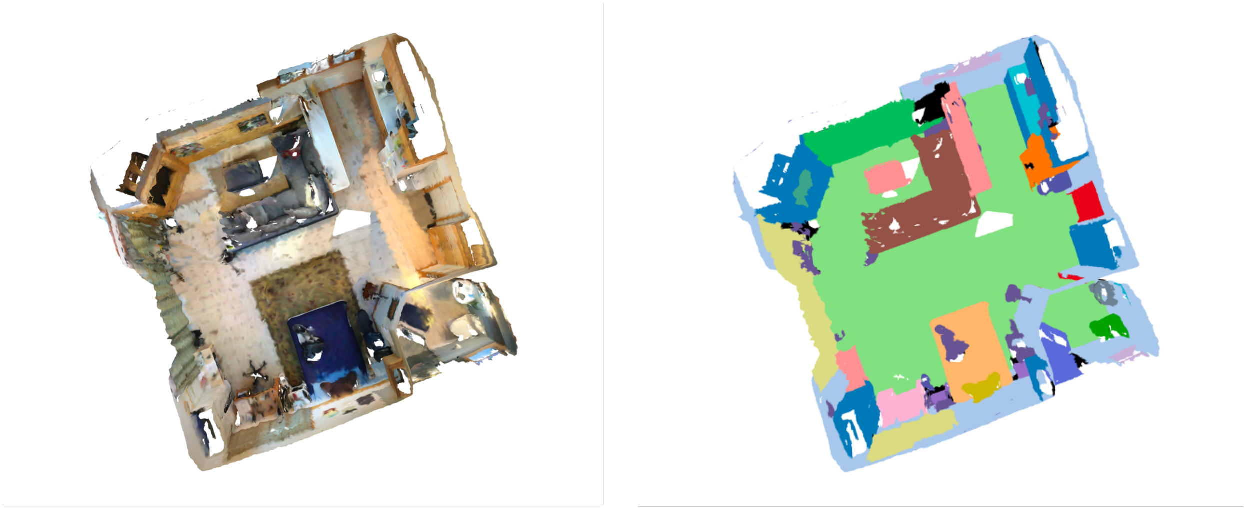
The process of scene selection involved careful consideration of the diversity and complexity of scenes to ensure a representative dataset. We aimed to include scenes with a wide variety of objects and arrangements to capture the variability encountered in real-world scenarios.

Figure 1. Visualization of the instance mesh of the bed belonging to scene0000_00 of ScanNet dataset.

B. Object Segmentation and Projection

For each selected scene, we processed the corresponding scan to project the mesh of each object into camera coordinates. This involved several detailed steps to ensure accurate segmentation and projection:

1) **Loading Scene Data**: We utilized Open3D, a powerful library designed for handling 3D data, to load the 3D mesh and segmentation data of each scene. This provided the foundational data required for further processing, enabling detailed manipulation and analysis of the 3D structures.
2) **Instance Mesh Creation**: From the complete scene mesh, we meticulously created instance meshes for



(a) 3D mesh indoor scene

(b) Segmented 3D mesh

Figure 2. Examples from scene0000_00 from ScanNet dataset: (a) A detailed 3D scan of an indoor scene, (b) Segmented 3D mesh highlighting individual objects.

each object. This step involved isolating the vertices and faces that corresponded to each individual object within the scene, ensuring that each object could be processed independently. Figure 1 shows an example of an instance mesh, highlighting the intricate detail captured in each segmentation.

3) **Projection and Mask Creation**: The mesh of each valid object was projected into the camera coordinates for each frame of the scan. This projection used the intrinsic and extrinsic camera parameters to create a 2D mask for each object, aiding in the segmentation of the object from the rest of the scene in each frame. This step was crucial for accurate representation and analysis of each object's features.
4) **Filtering Out Invalid Objects**: We applied a stringent filter to retain only those objects whose projection occupied at least 80% of the camera image. This threshold ensured that the objects were sufficiently visible in the frame, enhancing the accuracy and reliability of the segmentation and subsequent analysis.

Through these steps, we were able to isolate and project individual objects from different viewpoints within the scenes, creating a comprehensive dataset for training the image encoder of SigLIP to achieve multiview consistency. The before and after results are illustrated in Figure 3, showcasing the effectiveness of our method.

C. Frame Cropping and Object Isolation

Once we obtained the masks for the valid objects, several additional steps were necessary to finalize the dataset:

- **Isolating Object Frames**: We meticulously isolated frames where each valid object appeared. This involved iterating through each frame of the scan and applying the mask to segment the object. By doing so, we ensured that each frame contained a clear and unobstructed view of the object of interest.
- **Excluding Non-Useful Objects**: We excluded objects such as the floor, ceiling, and windows from this process, as they do not contribute to the multiview consistency task. This step was critical to focus our dataset on objects that are relevant for training the model.
- **Frame Cropping**: The frames were carefully cropped based on the masks to isolate the objects. These cropped frames were then saved as separate images for use in the fine-tuning process. This step ensured that the dataset was composed of high-quality images with clearly defined objects.

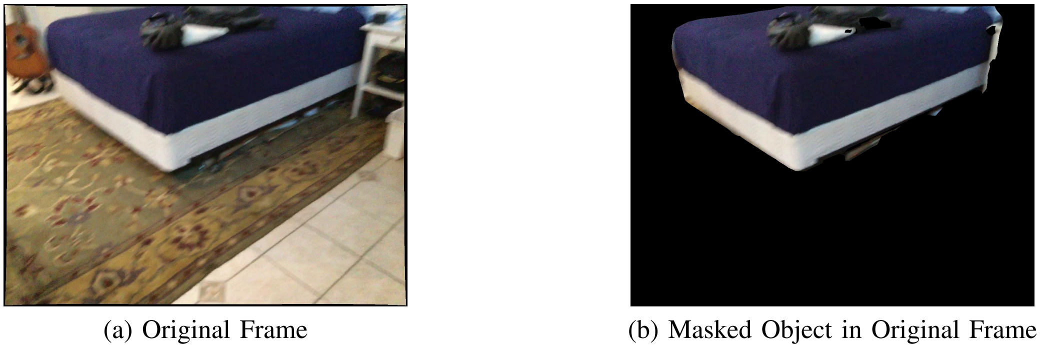
The before and after results are illustrated in Figure 4.

D. Ensuring Viewpoint Diversity

To ensure the dataset provided a variety of perspectives for each object, we implemented a strategy to select 15 images for each object. These images were uniformly distributed across the scan sequence. This uniform distribution guarantees that the images were captured from sufficiently different positions, providing diverse viewpoints of the same object.

The selection process involved:

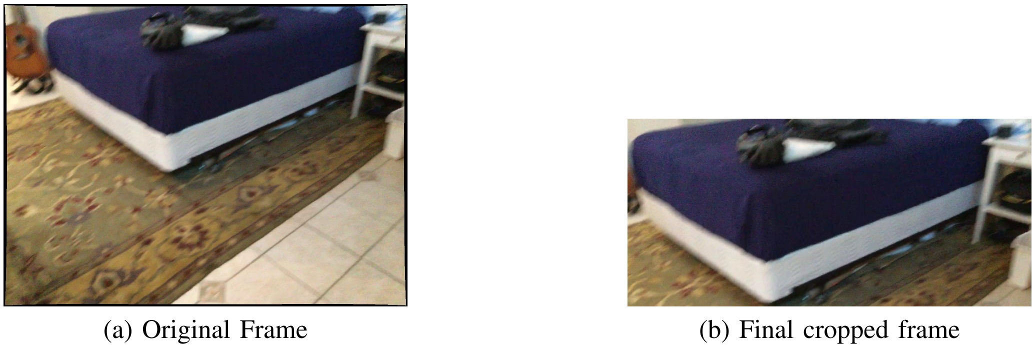
- **Uniform Distribution**: Ensuring that the selected frames were evenly spaced throughout the scan sequence to capture a wide range of angles and perspectives.
- **Maximizing Variability**: Choosing frames that offered the maximum variability in viewpoint to enhance the robustness of the dataset.



(a) Original Frame

(b) Masked Object in Original Frame

Figure 3. Visualization of the object segmentation and projection steps. The images represent frame 3498 of ScanNet scene000_00 before and after the segmentation and projection step: (a) Original frame from the scene, (b) Original frame with the background masked.



(a) Original Frame

(b) Final cropped frame

Figure 4. Visualization of the object segmentation and projection steps. The images represent frame 3498 of ScanNet scene000_00 before and after the dataset creation process: (a) Original frame from the scene, (b) Final cropped frame from the scene only including the specific object.

E. Avoiding Occlusion

To ensure that the objects were not occluded in the final images, we implemented a sophisticated method to filter out occluded vertices. This was done by comparing the depth coordinate of each projected vertex of the instance with the corresponding value in the depth map. If the depth from the depth map matched the depth coordinate of the vertex, the object was considered visible and not occluded. Conversely, if there was a discrepancy, the vertex was deemed occluded and was excluded from the mask creation. This approach ensured that only the visible parts of the objects were included in the final images, thus enhancing the accuracy and quality of the dataset.

Figure 5 demonstrates the importance of occlusion handling. Without accounting for occlusions, objects may be incorrectly segmented, as shown in the example where a wall occludes part of a bed. By excluding occluded vertices based on depth comparison, we ensure that only the visible portions of objects are included in the final masks, thereby improving the dataset's reliability.

This detailed process of dataset creation, including scene selection, object segmentation, frame cropping, viewpoint diversity, and occlusion handling, provided a robust foundation for training the image encoder of SigLIP to achieve multiview consistency. The comprehensive and meticulously prepared dataset is crucial for enhancing the model's performance in tasks requiring consistent feature representation across various viewpoints.

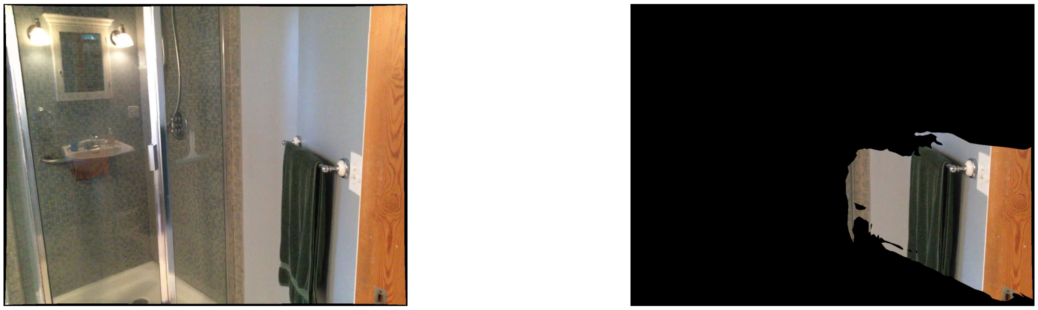
V. Initial Analysis

Prior to fine-tuning SigLIP, we conducted a thorough analysis of the image embeddings produced by the visual encoder of SigLIP. This analysis was essential to understand the baseline performance and multiview consistency of the embeddings.

The initial step involved extracting the embeddings for frames containing the same object viewed from different angles. Specifically, we focused on the embeddings representing a bed in scene0000_00 as a case study. The goal was to assess the inherent correlation between these embeddings before any fine-tuning was applied.

A. Cosine Similarity Analysis

To measure the consistency of the embeddings, we computed the cosine similarity between the embeddings of the frames. Cosine similarity is a common metric used to evaluate the similarity between two vectors, where a value of 1 indicates identical vectors and a value of -1 indicates completely dissimilar vectors. By calculating the cosine similarity for the embeddings of the same object



(a) Original Frame

(b) Masked Object in Original Frame

Figure 5. Effect of the occlusion handling method. These images represent frame 4009 of ScanNet scene000_00 before and after applying the occlusion handling method: (a) Original frame showing the bed partially occluded by a wall, (b) Masked object without considering occlusion, leading to incorrect segmentation.

from different viewpoints, we could quantify the degree of multiview consistency.

The resulting cosine similarity matrix provided a visual representation of the similarity scores between all pairs of frames. As shown in Figure 6, the matrix displayed some inherent correlation between the frames even before fine-tuning. This indicates that the embeddings produced by SigLIP's visual encoder had a baseline level of multiview consistency.

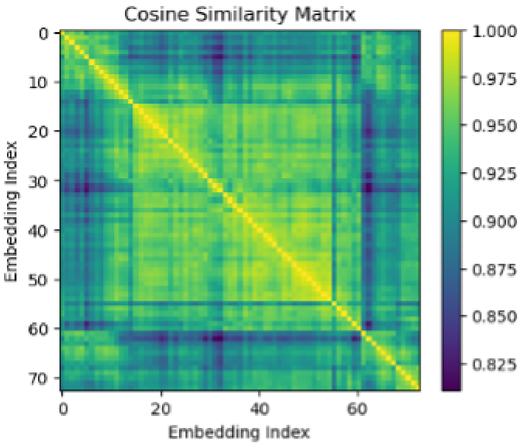


Figure 6. Cosine similarity matrix of embeddings representing a bed in scene0000_00 before fine-tuning.

B. K-means Clustering Analysis

To further explore the structure and consistency of the embeddings, we performed a K-means clustering analysis. K-means clustering is an unsupervised learning algorithm that partitions the data into K distinct clusters based on the features of the data points. For our analysis, we chose K based on the optimal number of clusters that could reveal meaningful groupings in the data.

The steps for the K-means analysis included:

1) **Determining the Number of Clusters**: We used the elbow method to identify the optimal number of clusters. This involved plotting the sum of squared distances from each point to its assigned cluster center and selecting the point where the rate of decrease sharply slows down (the ”elbow point”).
2) **Clustering the Embeddings**: Using the optimal number of clusters, we applied the K-means algorithm to partition the embeddings. Each embedding was assigned to one of the clusters based on its features.
3) **Evaluating Cluster Consistency**: We evaluated the resulting clusters to determine if embeddings from the same object and similar viewpoints were grouped together. High intra-cluster similarity (similarity within the same cluster) and low inter-cluster similarity (dissimilarity between different clusters) would indicate effective clustering performance. As shown in Figure 7, adjacent frames tend to be grouped within the same cluster. This clustering pattern occurs because the viewpoint changes gradually between close frames, resulting in similar images and thus similar embeddings.

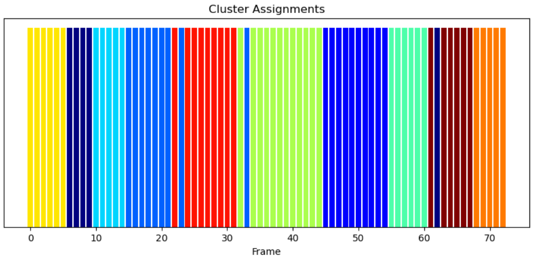


Figure 7. Cluster assignments of embeddings representing a bed in scene0000_00 before fine-tuning.

C. Principal Component Analysis (PCA)

To visualize the structure of the embeddings in a lower-dimensional space, we applied Principal Component Analysis (PCA). PCA is a dimensionality reduction technique that transforms the data into a new coordinate system, where the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

The PCA plot revealed the clustering of embeddings, showing how frames of the same object from different viewpoints were grouped together. This visualization helped us understand the intrinsic organization of the embeddings and the baseline level of clustering before fine-tuning. Images are reported and commented in section VI.

D. Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

In our analysis, we employed Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to identify clusters within the data. DBSCAN is a powerful clustering algorithm particularly well-suited for datasets with noise and clusters of varying shapes and sizes.

DBSCAN works by grouping together points that are closely packed (points with many nearby neighbors) and marking as outliers the points that lie alone in low-density regions (whose nearest neighbors are too far away). This method is advantageous for its ability to discover clusters of arbitrary shape without requiring a predefined number of clusters.

To determine the optimal value for the epsilon parameter (Eps), we evaluated the number of clusters and the number of noise points as functions of Eps. As shown in the left plot of Figure 8, the number of clusters varies with different values of Eps, while the right plot demonstrates how the number of noise points decreases with increasing Eps.

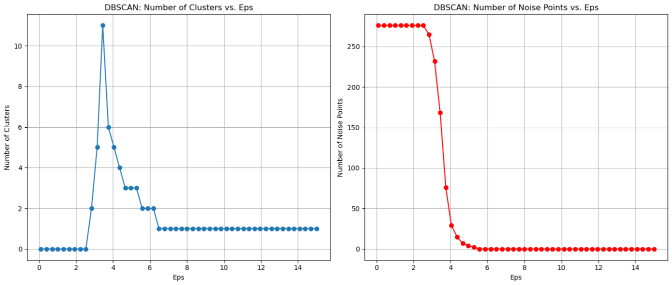


Figure 8. DBSCAN: Number of Clusters and Noise Points vs. Eps. The left plot shows the number of clusters identified as a function of Eps, while the right plot shows the number of noise points.

The DBSCAN analysis revealed distinct clusters within the data, highlighting regions of high density that correspond to significant patterns in the embeddings. This clustering helped us identify natural groupings in the data and provided insights into the underlying structure. The ability of DBSCAN to handle noise also allowed us to isolate and examine outliers, which could be crucial for refining our analysis and improving the robustness of our fine-tuning strategy.

E. Summary of Initial Findings

The initial analysis of SigLIP's image embeddings provided several key insights:

1) **Baseline Multiview Consistency**: The cosine similarity matrix revealed a baseline level of multiview consistency among the embeddings. Frames depicting the same object from different viewpoints exhibited inherent correlations, indicating the encoder's initial ability to maintain some degree of consistency.
2) **Variability in Embeddings**: The histogram of cosine similarity scores quantified the variability in the embeddings. This distribution offered a clear view of the range and concentration of similarity scores, reflecting the diversity of the embeddings' representations.
3) **Clustering Patterns**: The PCA and DBSCAN analyses highlighted distinct clustering patterns within the embeddings. These visualizations showed how frames of the same object were naturally grouped together in the feature space, emphasizing the encoder's capability to cluster similar embeddings.

These initial findings established a reference point for evaluating the impact of fine-tuning on the multiview consistency of the embeddings. By understanding the baseline performance, we could more effectively assess the improvements achieved through our fine-tuning approach.

VI. Fine-tuning SigLIP

A. Projection Head

To enhance the multiview consistency of SigLIP, we attached a projection head to the pretrained image encoder. The projection head consisted of several standard layers designed to refine the embeddings. The architecture of the projection head is shown in Figure 9, and it includes the following components:

- **Linear Projection Layer**: This layer, represented by ‘nn.Linear(embedding_dim, projection_dim)‘, maps the input embeddings to a higher-dimensional space. This transformation helps in capturing more complex relationships within the data.
- **GELU Activation Function**: The GELU (Gaussian Error Linear Unit) activation function, implemented as ‘nn.GELU()‘, introduces non-linearity into the model. This function is chosen for its smooth and differentiable properties, which help in better capturing the underlying data patterns compared to traditional ReLU.
- **Fully Connected Layer**: Another linear layer, ‘nn.Linear(projection_dim, projection_dim)‘, refines the projection further by transforming it within the new space. This step helps in adjusting the embeddings to be more discriminative.
- **Dropout Regularization**: The dropout layer, ‘nn.Dropout(dropout)‘, with a default dropout rate of 0.5, is used to prevent overfitting by randomly setting

a fraction of input units to zero during training. This helps in making the model more robust.

- **Residual Connection and Layer Normalization**: The residual connection (adding the input of the projection to its output before normalization) combined with layer normalization, ‘nn.LayerNorm(projection_dim, eps=1e-6)‘, stabilizes the learning process and ensures that the transformed embeddings maintain a consistent scale and distribution.

Figure 9 depicts the architecture of the projection head. This setup is intended to refine the initial embeddings, making them more suitable for tasks that require high multiview consistency.

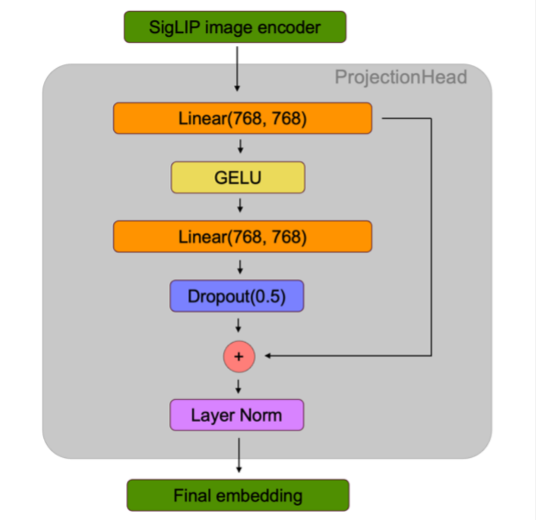


Figure 9. Architecture of the projection head attached to the SigLIP image encoder. The projection head includes linear layers, GELU activation, dropout regularization, and layer normalization to refine and enhance the embeddings.

B. Triplet Loss

The fine-tuning process employed a triplet loss function (see Figure 10). This loss function was designed to maximize the cosine similarity between embeddings of the same object (anchor and positive) while minimizing the cosine similarity between embeddings of different objects (anchor and negative). By doing so, the triplet loss ensured that embeddings of the same object were pulled closer together in the embedding space, while those of different objects were pushed further apart.

Figure 11 depicts the embeddings of seven distinct objects, with each color representing a different object and each point corresponding to the embedding of an object viewed from a specific angle, both before and after training. After just three epochs of training, the results are striking: seven distinct clusters have emerged. Embeddings of different objects have been effectively separated, while embeddings of the same object have been grouped closely together.

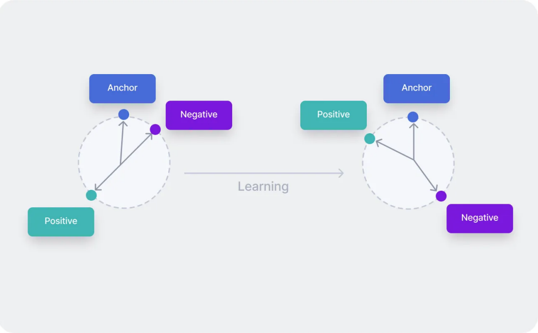


Figure 10. Visual representation of the triplet loss function process. The diagram shows how the anchor, positive, and negative samples are positioned relative to each other, demonstrating the goal of minimizing the distance between the anchor and positive while maximizing the distance between the anchor and negative.

C. Training Process

1) Data: The dataset comprised 350 scenes, each containing a variable number of objects. For each object, we selected 15 frames, equally spaced in time to capture significant changes in viewpoint. The dataset was divided into training and validation sets, with 80% of the scenes allocated for training and the remaining 20% for validation.

2) Configurations: We conducted experiments using two configurations: a single projection head and three concatenated projection heads. For all experiments, we utilized the Adam optimizer and a batch size of 512 triplets to ensure efficient and effective training. The two configurations of the projection head are shown in Figure 12.

D. Initial Experimentation

The initial experiment utilized only 10 scenes and a single projection head. Although this led to overfitting, it provided valuable insights into the behavior of the embeddings during training. The experiment was conducted with a batch size of 128 triplets and a learning rate of 0.001, using the Adam optimizer. Additionally, a scheduler with a step size of 5 and gamma of 0.5 was employed.

Over the course of 50 epochs, we monitored the PCA projections of the embeddings, which illustrated their movement during training (see Figure 13).

As Figure 14 shows, the training loss decreased to 0.15, while the validation loss exhibited oscillatory behavior with a minimum around 0.3. This experiment confirmed that the loss function was functioning as intended.

E. Full Dataset Training

After ensuring the loss function’s effectiveness, we applied the same workflow to the full dataset of 350 scenes. All the experiments used a batch size of 512 triplets to train the model efficiently and the Adam optimizer to update the model weights, ensuring convergence.

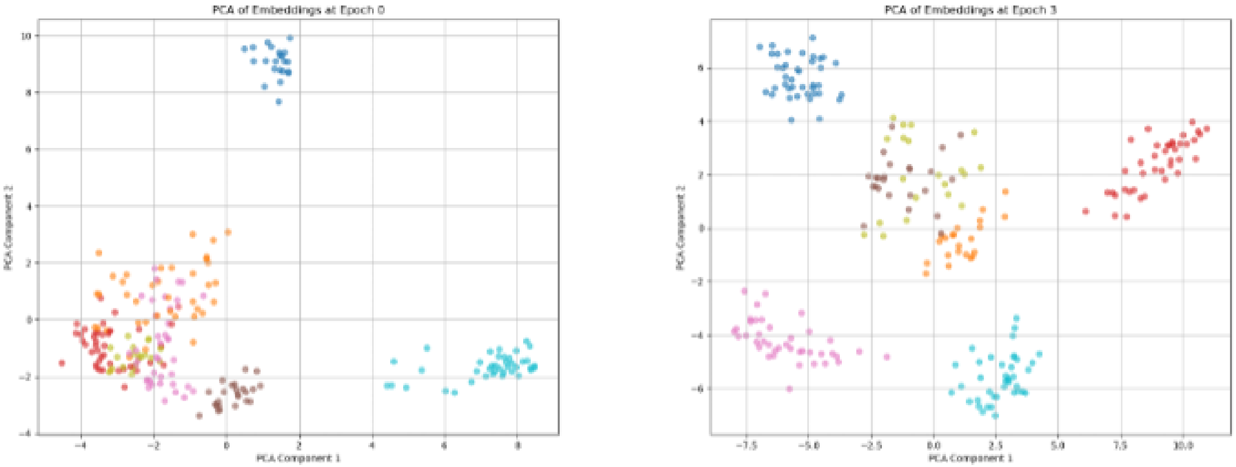


Figure 11. PCA of embeddings at Epoch 0 (left) and Epoch 3 (right). The images show the separation and clustering of embeddings representing seven distinct objects, each color-coded for a different object, after fine-tuning with the triplet loss function. The left plot shows the initial state, while the right plot shows the improved clustering after three epochs.

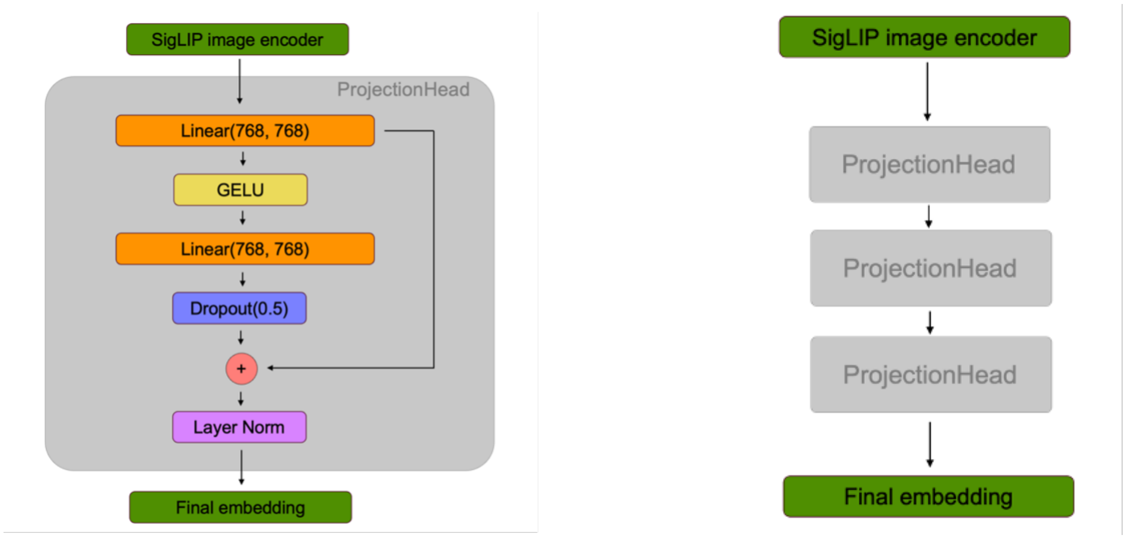


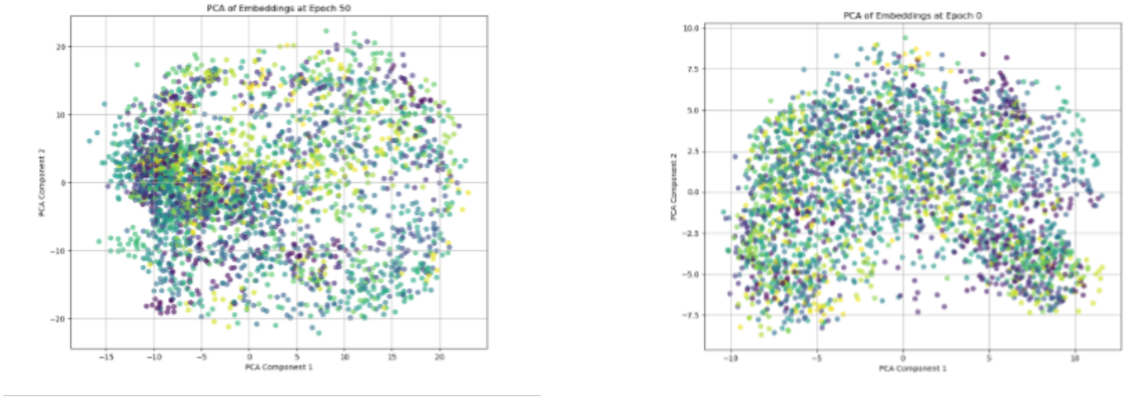
Figure 12. Configuration of the projection heads used in the experiments. Left: Single projection head consisting of a linear layer, GELU activation, another linear layer, dropout, and layer normalization. Right: Three concatenated projection heads to increase the complexity and capacity of the model for refining embeddings.

The training was performed using four different configurations:

- One projection head with learning rate 5e-5 without scheduler
- One projection head with learning rate 5e-4 with scheduler of step size 5 and gamma 0.5
- Three projection heads with learning rate 5e-5 without scheduler
- Three projection heads with learning rate 5e-4 with scheduler of step size 5 and gamma 0.5

As shown in Figure 15, the loss curves illustrate the training and validation loss over the epochs for the different configurations. The results provide insights into how the model's performance changes with different learning rates and the use of a scheduler. The *10-scenes* curve show the training and validation losses reported in Figure 14 for completeness.

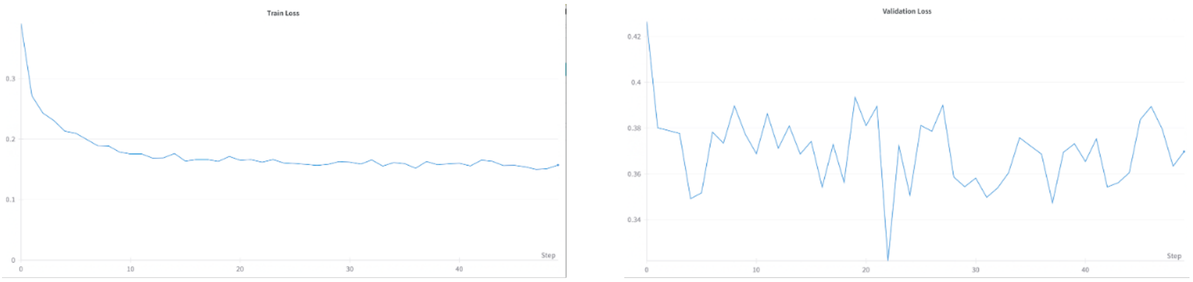
The fine-tuning of SigLIP using a projection head and triplet loss significantly enhanced the multiview consistency of the image embeddings. By applying this method to a larger dataset of 350 scenes, we demonstrated the scalability and robustness of the approach. This enhancement is crucial



(a) Embeddings before training

(b) Embeddings after training for 50 epochs

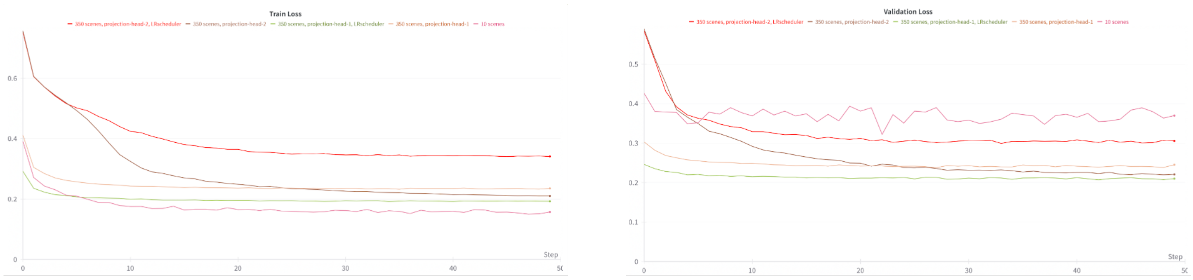
Figure 13. PCA projections of embeddings before and after 50 epochs of training.



(a) Training loss

(b) Validation loss

Figure 14. Loss curves for the initial experiment showing training and validation loss over epochs.



(a) Training loss

(b) Validation loss

Figure 15. Loss curves for the experiments showing training and validation loss over epochs.

for applications requiring consistent feature representation across multiple viewpoints, thereby improving the overall performance of SigLIP in tasks such as 3D object recognition and scene understanding.

VII. Future Work

In our current work, we have focused on fine-tuning the projection head attached to the SigLIP model. This approach has demonstrated promising results in enhancing the consistency of image and text embeddings. However, an intriguing direction for future research would be to explore the effects of fine-tuning SigLIP by unfreezing some of its internal layers.

Unfreezing internal layers allows the model to adapt more comprehensively to the training data, potentially leading to even better alignment and consistency between the visual and textual representations. This approach could further improve the model's ability to generalize across various tasks and datasets, providing deeper insights into the benefits and limitations of the SigLIP architecture when subjected to more extensive fine-tuning.

Investigating this direction would involve systematically unfreezing layers and fine-tuning them using a similar combined loss approach, possibly refining the balance between maintaining pre-trained knowledge and learning new, task-specific features. The potential improvements in embedding quality and model performance could pave the way for more robust and versatile applications of SigLIP in computer vision and natural language processing domains.

Moreover, we aim to enhance the text embeddings as well. The current strategy involves attaching a projection head to the pretrained text encoder of SigLIP and utilizing an additional dataset to fine-tune both the projection head applied to the image encoder and the projection head applied to the text encoder. The goal is to achieve similar results when comparing image embeddings and text embeddings before and after the projection heads are applied. An illustration of this is shown in Figure 16.

To ensure that the image embeddings retain their clustering property when representing the same object, we will employ a combined loss approach. The training process will alternate between the following two steps:

- **Triplet Loss for the Image Projection Head**: Using the custom dataset, we will apply the triplet loss to the image projection head. This step helps maintain the consistency and clustering of image embeddings for objects viewed from different perspectives.
- **Custom Loss for Text-Image Similarity**: We will implement a custom loss designed to preserve the relationship between the text and image embeddings before and after the projection heads are applied. This loss ensures that the alignment between the textual and visual representations remains coherent, even after the embeddings are transformed by the projection heads. To achieve this, we will use a dataset containing both annotations and images, such as the COCO dataset [8]. This step helps in fine-tuning the model to maintain the semantic relationships between images and their corresponding textual descriptions.

VIII. Conclusion

This report outlines the steps taken to enhance the multiview consistency of CLIP's feature representations using SigLIP. By creating a custom dataset from the ScanNet dataset and employing a triplet loss with a projection head, we demonstrated significant improvements in embedding consistency across different viewpoints. The ongoing work aims to further enhance the text embeddings to ensure a coherent relationship between textual and visual representations.

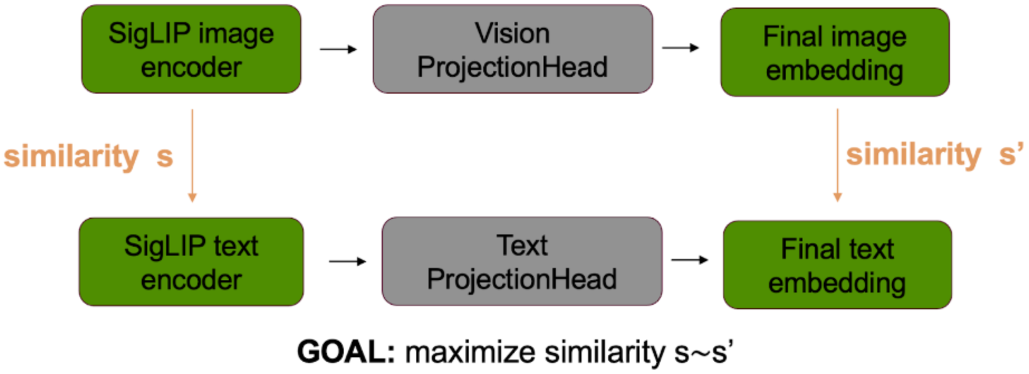


Figure 16. Schema illustrating the current approach to enhancing text embeddings alongside image embeddings.

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[2] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," 2023. [Online]. Available: https://arxiv.org/abs/2303.01469

[3] H. Zhang, Z. Pan, C. Zhang, L. Zhu, and X. Gao, "Texpainter: Generative mesh texturing with multi-view consistency," 2024. [Online]. Available: https://arxiv.org/abs/2406.18539

[4] C. Zhang, J. Tong, T. J. Lin, C. Nguyen, and H. Li, "Pmvc: Promoting multi-view consistency for 3d scene reconstruction," in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024, pp. 3666–3676.

[5] M. E. Banani, A. Raj, K.-K. Maninis, A. Kar, Y. Li, M. Rubinstein, D. Sun, L. Guibas, J. Johnson, and V. Jampani, "Probing the 3d awareness of visual foundation models," 2024. [Online]. Available: https://arxiv.org/abs/2404.08636

[6] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," 2023. [Online]. Available: https://arxiv.org/abs/2303.15343

[7] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," 2017. [Online]. Available: https://arxiv.org/abs/1702.04405

[8] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015. [Online]. Available: https://arxiv.org/abs/1405.0312