POLITECNICO DI TORINO

MASTER's Degree in DATA SCIENCE AND ENGINEERING





MASTER's Degree Thesis

Decomposing the Modality Gap: Single-Axis Attribution of Representation Geometry in Multimodal Large Language Models

Supervisors

Prof. Giuseppe RIZZO

Dr. Federico D'ASARO

Dr. Luca CATALANO

Candidate

Hassine EL GHAZEL

SEPTEMBER 2026

Summary

Connector-based vision-language models link a frozen visual encoder to a pretrained LLM through a trainable projector. In the shared embedding space, image and text embeddings occupy separate regions. This separation is called the modality gap, and early work summarized it as the distance between the centroids of the two modality clouds.

Previous research measures this discrepancy inside a contrastive dual encoder, reduces it to a single number, and corrects it with one fixed transformation, reporting gains on retrieval and zero-shot classification. However, we consider this phenomenon in terms of an MLLM in image captioning, where a pretrained decoder must generate a caption from a visual representation. Our contribution has two aspects. First, we explore this divergence in contrast to retrieval, where we measure the gap with respect to the generated caption. Second, we treat this gap not as a single distance but as a composite concept across different axes, manipulating them one by one to analyze their contribution to caption generation. In our case, the gap consists of four measurable components: the distance between the centroids, the total variance of the image cloud, the number of directions in which this variance is distributed, and the alignment of the subspaces.

A diagnostic model is trained to move one axis at a time while holding the others fixed, and its effect on the quality of captions is interpreted against a baseline in which no axis is driven.

The effects are notably asymmetrical. Reducing the centroid distance improves caption quality on all metrics used, increasing reference-free image-text agreement while decreasing hallucinations and raising object recall. In contrast, compressing variance degrades all of them. Redistributing that variance over fewer directions leaves caption quality unchanged. Orientation, the axis a contrastive objective controls most directly, is the easiest to adjust but returns at most a third of what the centroid distance returns.

Overall, decreasing the centroid distance is a real lever on how well a decoder with frozen weights and low-rank adapters describes what it is shown.

Acknowledgements

This work was carried out at Fondazione LINKS, whose computing infrastructure made it possible to train, measure and discard a great many models on the way to the few that were used.

I am most grateful to Dr. Federico D'Asaro and Dr. Luca Catalano, who followed this project closely and consistently asked the inconvenient question rather than the easy one. I finish this work more precise and more attentive to detail than I began it, and that is their doing.

I thank prof. Giuseppe Rizzo for supervising this thesis and for the opportunity to develop it at LINKS.

Table of Contents

Summary **I**

Acknowledgements **II**

1 Introduction **1**
1.1 Outline of the thesis 2

2 Related Work **4**
2.1 Vision-language models 4
2.1.1 Architectures 4
2.1.2 Contrastive pre-training and CLIP 6
2.1.3 Multimodal large language models 7
2.2 Geometric properties of the embedding space 8
2.2.1 The modality gap 8
2.2.2 Strategies for reducing the gap 10
2.3 Image captioning 14
2.3.1 Evaluation metrics 14
2.3.2 Object hallucination 14

3 Methodology **16**
3.1 Architecture 16
3.2 Training protocol 17
3.3 Measuring the gap 19
3.3.1 The proposed four axes 19
3.3.2 Reduction of the diagnostic suite 20
3.4 Control must act on the measured object 22
3.5 Geometry objectives 23
3.6 The proposed objective: closing the location gap 25

4 Experiments **26**
4.1 Computational budget 26
4.2 Datasets and evaluation metrics 26
4.3 The conditions 29
4.4 Main result 30
4.5 Ablations 31

4.5.1 Inter-Metric Agreement in Condition Ranking 32
4.5.2 Structural Origins of the Modality Gap 33
4.5.3 Performance Saturation Under Contrastive Scaling 34
4.5.4 Objective Targeting: [CLS] versus Pooled Representations . . 35
4.6 Factorial Analysis of Geometric Interventions 36
4.6.1 Training-time verification 39
4.6.2 Qualitative examples 42
4.6.3 Location dose–response 48
4.7 Ablating the Alignment Target: Structural vs. Semantic Anchors . . 48
4.8 Distance to a fully-resourced reference 50
4.8.1 The reference 50
4.8.2 How far a single-axis move gets 50
4.8.3 Scaling the location drive 51
4.8.4 The geometry of the remaining distance 54
4.8.5 Qualitative examples along the location ladder 55

5 Conclusions **59**
5.1 What was established 59
5.2 Limitations 60
5.3 Future directions 60

A Implementations **61**
A.1 The connector 61
A.2 The geometry objective 61
A.3 Measuring the orientation axis 63

Bibliography **64**

List of Tables

3.1 Model configuration. 17

3.2 The two stages side by side. They differ in objective, in where the loss is computed, and in what is allowed to move. The optimiser settings that implement them are experimental detail and are reported in Table 4.1. 18

3.3 The complete diagnostic suite, measured on MSCOCO `val2017`, n=5000, d=4096, over the eight conditions `C1_stage2`, `C2_stage1`, `C3_stage2`, `C3pinr`, `Cloc`, `Corient`, `Crank15` and `Cscale1500`. Only the first block satisfies all three requirements and defines the factorial design; the reason for each exclusion is given in the text. 22

3.4 The method of Equation (3.9). One axis is driven, two are held, and all four terms act on the pooled-257 vector. 25

4.1 Optimiser settings. Both stages are described in Section 3.2 and compared in Table 3.2; all conditions share these settings and differ only in the geometry term added to the Stage-2 objective (Section 3.5). 26

4.2 Datasets used for training and evaluation. Stage 2 includes examples from `train2017` images disjoint from the evaluation dataset `val2017`. 27

4.3 The two evaluation regimes. The first checks for brief responses to the provided image, while the second focuses on a detailed description decoding up to 256 tokens. The metric sets differ by design: BLEU, CIDEr, METEOR and SPICE compare against one-sentence MSCOCO references and lose their meaning at four times that length, so the detailed regime uses reference-free measures only. The gap statistics of Section 3.3 belong to neither regime, being computed from the projected embeddings of the same 5,000 images without decoding. . 27

4.4 The downstream metrics, their defining equations and ranges. An up arrow marks a metric where higher is better, a down arrow where lower is better. 29

4.5 The training conditions. The first block separates the contribution of each training stage, the second uses a contrastive objective, the third is single-axis factorial, where one axis is changed and the other three are kept fixed and measured, and the fourth gives the two dose-ladder rungs that appear alongside the factorial conditions in Table 4.7. . . 30

4.6 The method (Cloc) against the geometrically pinned baseline, with the combined variant (Clocorient) for reference. CLIP is CLIPScore, CH is CHAIR, rec. is object recall, obj/cap is objects mentioned per caption; an up arrow marks a column where higher is better, and the best value per column is in bold. The method gains +0.0526 CLIPScore on the brief set, worth 22.9 standard errors, and all three of its CHAIR differences have 95% bootstrap intervals excluding the baseline. The two n-gram metrics sit near their floor on captioners trained this briefly, so they separate the conditions by very little: the method holds the best of both, and CIDEr clears 10^{-4} for no other condition. Caption length is in Table 4.16. 31

4.7 All eight conditions on both evaluation sets, ordered by brief-set CLIPScore, which is also the order every one of the four ranking measures produces. CLIP is CLIPScore, CH is CHAIR, rec. is object recall; an up arrow marks a column where higher is better, and the best value per column is in bold. The two n-gram columns are the exception to the common order: both are maximised by the dose-ladder rung rather than by Clocorient, they separate the eight conditions over a range of 0.0150 in BLEU-4, and CIDEr clears 10^{-4} for only the two strongest location drives. The conditions are defined in Table 4.5. . . 33

4.8 Gap geometry by training stage, MSCOCO `val2017`, n=5000. Chance-level overlap at q=16 is $q/d \approx 0.0039$, from Equation (3.4). 33

4.9 Captioning by training stage, n=5000. The three degenerate conditions were scored with n-gram metrics only, so their reference-free and scene-graph entries are unavailable. 34

4.10 full+orient: contrastive dose sweep, n=5000. CLIP is CLIPScore. Every geometric quantity moves monotonically with λ; the quality metrics do not. BLEU-4 makes the point sharply: it peaks at $\lambda = 0.7$ and then falls to 0.0453 at $\lambda = 0.9$, below the 0.0455 of the run with no contrastive term at all, so the strongest dose of the objective that moves orientation furthest is the one that buys nothing. Effective rank rises from 38.61 to 42.56 across the sweep and is omitted for width. . 35

4.11 The same objective applied to two different vectors, n=5000. "pin" is the scale pin; CLIP is CLIPScore. 36

4.12 Single-axis isolation, n=5000. Bold entries are the targeted axis; the other columns move by at most 16% against on-axis moves of 45% to 116%, so the isolation is verified from measurements rather than assumed. 36

4.13 Effect of each intervention on both evaluation sets. CLIP is CLIPScore, CH is CHAIR, rec. is object recall. z is against all-pinned on the brief-set CLIPScore; an asterisk marks a CHAIR value whose 95% bootstrap interval excludes the baseline. Lower CHAIR is better, higher everything else. The two n-gram columns agree with the rest on which conditions help and which hurt, but over a far narrower range: BLEU-4 spans 0.0313 to 0.0445 across all six, and CIDEr stays below 10^{-4} everywhere except Cloc. 37
4.14 Location dose–response, n=5000. z is against the preceding row. The scale pin is active in the lower three conditions, holding variance within 11%. 48
4.15 Alignment-anchor ablation, dd256, n=1300. z is against the corresponding embedding-table condition. 50
4.16 Comparison against a fully-resourced reference, dd256. A bold entry marks the best of our conditions in each column; lower CHAIR is better. Steps are adapter steps at an effective batch of 32, so the rungs of the location ladder see 14,400, 48,928 and 80,000 items. The reference is evaluated for inference only, and the reference-based metrics are absent here because the detailed regime scores 256-token descriptions against one-sentence references, as Table 4.3 sets out. 50

List of Figures

2.1 A typical multimodal large language model. (Top) A modality encoder turns the input into features, a connector maps them into the input space of the language model, and the LLM produces text; the optional generator is used when the model must emit modalities other than text. (Bottom) The three connector interfaces: a projection-based MLP as in LLaVA, a query-based Q-Former as in BLIP-2, and a fusion-based connector that operates inside the attention of the LLM. Source: [14]. 6

2.2 Contrastive pre-training in CLIP. Within a batch of N pairs, the matched image-caption products on the diagonal are the positives and every off-diagonal product is a negative. Source: [3]. 7

2.3 The LLaVA architecture. A frozen vision encoder maps the image $\mathbf{X}_v$ to features $\mathbf{Z}_v$, a single projection $\mathbf{W}$ maps those to $\mathbf{H}_v$ in the same input embedding space that the instruction $\mathbf{X}_q$ occupies as $\mathbf{H}_q$, and the language model f_ϕ attends over the concatenated sequence to emit the response $\mathbf{X}_a$. Source: [1]. 8

2.4 The modality gap in multimodal contrastive learning. (a) Two encoders map each modality onto the hypersphere under a symmetric contrastive loss. (b) UMAP of paired embeddings from pre-trained models, where lines join matched pairs: the two modalities occupy disjoint regions across natural images, video, medical images and amino-acid sequences. (c) The same architectures with random weights, showing the gap is present before any training. Source: [4]. 9

2.5 Contrastive learning preserves the gap. (a) Both modalities are shifted towards each other and renormalised. (b) At CLIP's learned temperature $\tau = 1/100$ the validation loss is minimised at the separation the model actually settles at, marked by the dashed line, so closing the gap makes the loss worse. (c) and (d) The minimum moves away from that separation as the temperature rises, and the structure disappears at $\tau = 1$. Source: [4]. 10

2.6 The ReAlign pipeline. Anchor Alignment translates the source modality onto the target centroid, Trace Alignment matches total residual energy by a single scalar, and Centroid Alignment removes the drift that spherical renormalisation reintroduces. Source: [6]. 12

2.7 Evidence for an anisotropic residual. (a) Translating the text embeddings onto the image centroid leaves 89% of the discrepancy in place. (b) The residual spectrum departs from the isotropic baseline $1/d$. (c) Residual energy concentrates in a low-dimensional subspace, with $A_r = 28.6$ and $d_{\text{eff}}/d = 0.284$. Source: [7]. 13

3.1 The model architecture. The connector is the only module that is trained in both stages, and its output is an input to the LLM and is where the gap was measured. All the gap metrics were computed in float64. 17

3.2 The two training stages. Stage 1 uses only the LLM's embedding lookup table. Stage 2 uses a loss computed only on the response tokens, and the connector receives gradients exclusively via its effect on the generated text. 19

3.3 The five training conditions as geometric interventions on the image cloud. In each panel the dashed ellipse is the cloud before the intervention and the solid one after it, while the text cloud Y is fixed throughout because it comes from the frozen embedding table. Every condition is Equation (3.7) with the $\mathcal{L}_{\text{geo}}$ named beneath its panel, and the three axes it does not drive are held at their post-Stage-1 setpoints by the pins of Equation (3.6). Panel (a) is the baseline: all four axes are pinned, only the language objective is free, and it therefore carries no dashed outline. In (d) and (e) the pin setpoint is moved rather than frozen, which is what turns a pin into a drive. The conditions are named in Table 4.5. 24

4.1 Where each intervention leaves the image cloud. Each panel holds the pooled image embeddings of one condition, n=1300, projected onto that condition's own gap direction (horizontal) and the leading direction of its variance orthogonal to it (vertical). The red star is the text cloud, which is identical in every panel because the text tower is frozen, and the dashed arrow is the centroid distance G_μ. Scale, shape and orientation leave the cloud where the baseline put it, at 177.6 to 177.9; only Cloc and Clocorient move it, to 30.97 and 11.72. Cscale1500's cloud is visibly smaller, which is its $\operatorname{tr}\Sigma_X$ of 1487.6 against roughly 4300 elsewhere. Shading is a kernel density estimate over the same points. 37

4.2 Location. Cloc and Clocorient log the gap to the text centroid directly and drive it from 177.9 to 32.8 and 14.8. The four conditions that pin location log only displacement from their own frozen centroid, so their gap is not measured; the band is the interval the triangle inequality allows, 177.86 plus or minus that displacement, and it never approaches the two driven conditions. 40

4.3 Scale, as total variance divided by the common baseline $\operatorname{tr}\Sigma_0 = 4767.3$, auto-measured at the first step. A pinned axis sits on 1 and the shaded band is ±10%. Five conditions hold there; only Cscale1500 leaves, settling at 0.32, which is its target of 1500 against that baseline. The quantity is reconstructed from the squared pin term, which loses the sign of the deviation, so it is drawn as a band rather than a line. . . 40

4.4 Shape, as the batch participation ratio divided by the common baseline $r_0 = 23.30$. Five conditions hold at 1; only Crank15 leaves, settling at 0.64, which is its target of 15 against that baseline. Cscale1500's excursion over the first hundred steps returns to the band. This ratio is logged directly, so no reconstruction is involved. 41

4.5 Orientation, as the contrastive term the orientation drive optimises, with chance for a 64-image batch marked at $\ln 64$. The conditions that pin orientation sit just under chance and stay there; Corient moves it to 3.63 and Clocorient, which carries the same dosage alongside the location drive, to 1.88. Cloc is absent because the distance trainer has no contrastive leg and never evaluates this term; its evaluated overlap is at baseline, 0.0519 against 0.0515, so orientation is untouched there. 41

4.6 The autoregressive loss over the same runs, on a logarithmic axis, per step in light and as a 15-step rolling median in bold. Every condition falls from its initial value and settles together, so none of the geometric terms is bought by degrading the language objective. 42

4.7 A sandwich and fries. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The baseline and the shape drive both describe a man holding a furry dog in a kitchen, spending nine and eight words on objects that are not there, and the scale and orientation drives open on a fixed fragment. Both driven conditions describe the plate itself, a sandwich on a bun beside a portion of fries, and invent nothing. Cloc reaches 0.859 against the baseline's 0.471, the largest gain in these examples. 43

4.8 Zebras grazing. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. Every condition that leaves the gap open calls the animals horses, and compressing the scale abandons the scene for a living room. Both driven conditions name the zebras and score above 0.82, against 0.56 for the baseline, the largest margin in these examples. 44

4.9 A clock tower. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The captions run longer here and are set smaller so that each is shown in full. The baseline, the scale drive and the orientation drive all move the scene indoors, spending 13, 14 and 11 words on objects that are not there, and only the two driven conditions name the clock. The shape drive is the cautionary case: its caption is wrong throughout, a decorated house rather than a tower, yet it counts no hallucinated object, because house, wreath and Christmas tree are not annotated categories, and only the reference-free score registers the error. 45

4.10 Reading at a desk. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The baseline describes an empty room and misses the person entirely. Both driven conditions produce the cleanest captions in this set, a man seated at a desk reading, with no invented object and no invented count, and they score 0.78 against the baseline's 0.63. 46

4.11 A polar bear on a rock. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The single annotated object is the bear, and no condition that leaves the gap open recovers it: two see a dog on a dock, one a cat on a couch indoors and one only people beside the water. Both driven conditions name the bear, although neither reads the setting correctly, one adding snow and the other a river with onlookers. 47

4.12 The location ladder against training volume, dd256 with n=1300. All four measures are dimensionless rates, so they share one vertical axis but are read in their own units; the arrow beside each name gives the direction of improvement. Solid segments join the three Cloc checkpoints and a dashed segment carries each metric across to its star, the LLaVA-1.5-7B reference. The horizontal axis is broken: the ladder spans 14.4k to 80k items and the reference sits at 1.22M, so a continuous axis would give one point more width than the whole ladder. The dashed segments therefore interpolate across the period the model was trained between 80k and 1.22M items. The item counts are those each location-drive run consumed on top of the two-stage baseline that every condition shares, against the reference's full training, so the two are comparable in order of magnitude rather than exactly. CLIPScore reaches 0.7033 against the reference's 0.7910 and recall 0.6675 against 0.7738, while the distance left on CHAIR_s is far larger. 52

4.13 The location gap during training. The light trace is the raw loss on the training batch and the solid blue line its 25-step rolling median. The three coloured markers are the full-set G_μ at the checkpoints where it was saved and evaluated, measured over 5000 pairs rather than one batch of 32, which is why they do not sit exactly on the batch curve. The vertical line at 1529 marks the warm restart [45] where Cloc_80k resumes Cloc_long 53

4.14 The autoregressive loss over the same 2500 steps on a logarithmic axis, per step in light and as a 25-step rolling median in bold. It falls from 11.2 at step 1 to about 1.3 within the first hundred steps, and the hundred-step means centred on steps 450, 1050, 1500 and 2450 are 1.175, 1.126, 1.123 and 1.130. 53

4.15 The two pinned statistics, each as a ratio to its own target, $\operatorname{tr}\Sigma_0 = 4767.3$ for scale and $r_0 = 23.31$ for shape, measured on the training batch. The shaded band is $\pm 10\%$. The opening dip is the pins taking hold over the first 150 steps; afterwards they track their targets, within 2% on average over the whole schedule and within 0.5% over the continuation. 54

4.16 Teddy bears on a bed. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bear, bed, teddy bear. 56

4.17 A cat asleep on a bed beside a bookshelf. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bed, book, cat, laptop, teddy bear. 57
4.18 A bathroom with a sink and a toilet. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bottle, sink, toilet. 58

Acronyms

MLLM	Multimodal Large Language Model.
LLM	Large Language Model.
CLIP	Contrastive Language-Image Pre-training.
UMAP	Uniform Manifold Approximation and Projection.
ViT	Vision Transformer.
MLP	Multi-Layer Perceptron.
GELU	Gaussian Error Linear Unit.
LoRA	Low-Rank Adaptation.
NF4	4-bit NormalFloat.
AR	Autoregressive.
InfoNCE	Information Noise-Contrastive Estimation.
MSCOCO	Microsoft Common Objects in Context.
CHAIR	Caption Hallucination Assessment with Image Relevance.
SPICE	Semantic Propositional Image Caption Evaluation.
CIDEr	Consensus-based Image Description Evaluation.
BLEU	Bilingual Evaluation Understudy.

METEOR Metric for Evaluation of Translation with Explicit ORdering.

POPE Polling-based Object Probing Evaluation.

Chapter 1

Introduction

Multimodal large language models (MLLMs) have become a standard method for giving language models vision capabilities. Rather than training such a network from scratch, MLLMs keep the pre-trained visual encoder and the language decoder frozen, instead training a small bridging module between them [1][2]. This module translates visual features into the token embedding space expected by the decoder. The design of late-fusion systems like these is inexpensive and powers most existing open-source models. They also centralize cross-modal understanding to a single module, with only the connector bridging visual input to text output.

The connector must solve a problem not yet mastered by contrastive models. On dual encoder models like CLIP [3], visual and caption embeddings occupy separate regions of the joint embedding space. This is referred to as the modality gap. Both initialization by virtue of the cone effect, as well as the contrastive training objective widen the gap [4]. For late-fusion models, the output of the connector is piped directly into the decoder input space. If the connector produces embeddings outside the distribution encountered during pre-training, the decoder must process inputs that differ from those it was trained to interpret.

There are three limitations in how prior work has studied this gap. First, it was measured as a single number: how far apart the centers of each modality's embedding cluster are [4]. Clusters may differ in center, but also size, isotropy of that variance across dimensions, and rotation of those dimensions with respect to one another. Second, efforts to close the gap typically apply a single transformation to frozen embeddings which shifts many properties at once, then only report total effect [5][6][7]. This multiplicity prevents analysis of what exact contribution caused improvement. Third, there is some debate over which axis matters most. Recent work hints that centroid distance may be negligible, with the gap being dominated by an anisotropic residual in a small number of directions [7]. Of course, this is all dependent on task. These quantities are all measured with respect to normalized dual encoder space via inner products, but the model studied in this thesis must generate a sentence.

This thesis investigates which properties of the modality gap a frozen language decoder is most sensitive to when performing image captioning within an MLLM. We

frame the gap as consisting of four measurable axes: centroid distance, total variance of the image cloud, dimensionality over which that variance is spread, and alignment of principal subspaces. We selected these four axes by evaluating twenty candidate statistics against three selection criteria.

Measuring the utility of this decomposition relies on a crucial property of the proposed method. Modifying weighting of a contrastive alignment term causes each axis to vary monotonically, but in a manner tightly coupled enough that it does not allow cleanly attributing effects to individual axes. To decouple them, we train under a factorial design where each condition pins every axis except one to its baseline value. Naturally, these pinned axes must be verified to have held during training. Additional methodology is taken to keep tests fair: all conditions act on the same pooled vector that the metrics report and the decoder consumes, with matched training hyperparameters, meaning that any difference in outcome is caused exclusively by the added term.

The contributions of this thesis are as follows.

- A study of the effect of modality gap, understood internally to a multimodal large language model tasked with image captioning, where the gap is measured against the text the model generates rather than against what it retrieves.
- A decomposition of the gap into four reasonably non-overlapping axes which can be independently controlled by a differentiable penalty operating on a single batch.
- An axis-wise attribution of caption quality to each axis in turn, obtained by training conditions where one axis is driven while the other three are pinned. We show that centroid distance is the effective component, and that the axis the vanilla contrastive objective attempts to optimize is likely the least important of the three that have an impact on quality.
- A training objective constructed from that attribution alone, applied under the axis of location while keeping the others fixed, which demonstrates improved performance over an otherwise identical model with the geometric terms pinned.

1.1 Outline of the thesis

This thesis is organized as follows. Chapter 2 reviews related background information. We start by discussing vision-language models in general and introduce the contrastive objective that leads to a joint embedding space. Next, we cover existing work that critiques that space and proposed solutions to narrow the gap, concluding with introduction to the task of image captioning and evaluation metrics. Chapter 3 details our model architecture and training procedure. There we describe the proposed four axes, motivation for choosing them, and how they are computed via geometric objectives into a single loss. Chapter 4 presents our experimental evaluation of said method compared to a geometrically pinned baseline, ablation studies that

confirm our attribution is correct, and comparison to a fully-resourced reference model. Finally we discuss limitations and future work in Chapter 5 and conclude.

Chapter 2

Related Work

In this chapter, we will provide an overview of existing literature most relevant to this thesis. We begin in Section 2.1 with an overview of vision-language models. We discuss the two families of architectures in use, the contrastive objective under which they learn a shared embedding space, and multimodal large language models, which place a decoder on top of a visual encoder. We then discuss the geometric properties of the embedding space in Section 2.2. We start with the modality gap and its explanation, then continue with techniques proposed to reduce it. We conclude this literature review by introducing image captioning, the task on which this thesis is evaluated, in Section 2.3 alongside the metric families used to score captions and object hallucination specifically.

2.1 Vision-language models

Vision-language models have emerged as the standard paradigm for bridging visual data and natural language. By training on datasets composed of image-caption pairs, they learn representations in which an image and a description of it can be compared directly. The use made of that representation depends on the architecture: a dual encoder scores image-text pairs, which supports retrieval and zero-shot classification without further training, whereas a model built around a language decoder generates the description itself. The way the two modalities are joined also determines where those representations can be extracted, and therefore what can be measured. We start with a look at the architectures.

2.1.1 Architectures

Three families are in common use, and this thesis belongs to the second of them.

- **Dual encoders (two-tower models)** use one encoder per modality and map both into a single space of fixed dimension. Everything the model does afterwards is an inner product between two vectors in that space, which makes retrieval scalable because embeddings can be precomputed. Both towers are

trained, so both sides of the comparison are free to move. This family includes CLIP [3] and ALIGN [8].

- **Connector-based models** keep a pre-trained visual encoder frozen, pass its output through a small trainable module, and hand the result to a pre-trained language decoder. They differ in the interface that module presents. A two-layer MLP projects the visual features into the decoder's input space in LLaVA [1][9]; a Q-Former compresses them into a fixed set of learned query tokens in BLIP-2 [2]; and gated cross-attention injects them into the decoder's internal layers, rather than at its input, in Flamingo [10].

- **Unified models** remove the separate visual encoder altogether. Images are tokenized into a vocabulary shared with text and a single network is trained over both modalities from the start, so there is no connector and no point at which two separately learned representations must be joined. Chameleon [11] is the canonical open example.

This grouping is not exhaustive. Single-stream transformers such as ViLT [12] and UNITER [13] concatenate patch and word embeddings into a single transformer without a shared discrete vocabulary, and belong to the generation that preceded large language decoders. Models whose output is an image rather than text are outside the scope of this review, which is concerned with representations that are read out as language.

Naming for the second family is not settled. Work contrasting it with the third calls it late fusion [11]. This thesis avoids that term, since in the older sense of the word it is the two-tower design whose streams stay apart until a single score at the end, and uses connector-based throughout.

Figure 2.1 places the three connector interfaces inside the full architecture.

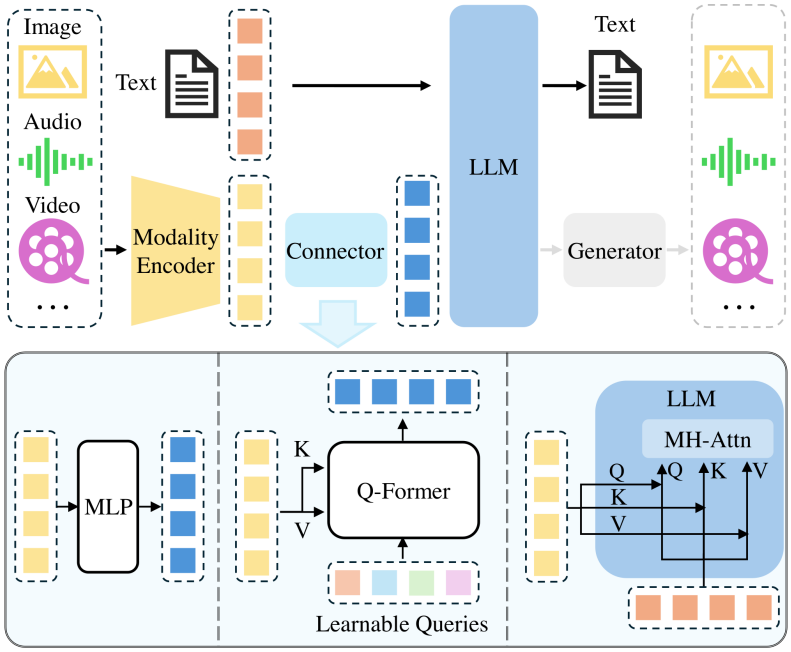


Figure 2.1: A typical multimodal large language model. (Top) A modality encoder turns the input into features, a connector maps them into the input space of the language model, and the LLM produces text; the optional generator is used when the model must emit modalities other than text. (Bottom) The three connector interfaces: a projection-based MLP as in LLaVA, a query-based Q-Former as in BLIP-2, and a fusion-based connector that operates inside the attention of the LLM. Source: [14].

This difference matters for the upcoming reasoning. In a dual-encoder setup, both modalities are trainable, so the geometric characteristics of the shared space result from two moving entities. In contrast, in a connector-based model the image reaches the decoder only through the connector. It is therefore both the only place where the visual representation can be altered and the only point where the two modalities are mapped to the same space and can be directly compared, which allows control of the underlying geometry in the first place. The architecture used in this thesis is introduced in Section 3.1.

2.1.2 Contrastive pre-training and CLIP

CLIP, introduced by Radford et al. [3], trains its two towers jointly on a large dataset of web image-text pairs. This is done by minimising a symmetric InfoNCE loss [15]. Given a batch, the embeddings from the matched image-caption pair are considered positive and all other combinations negative. The similarity that enters the softmax is the cosine similarity of the embeddings divided by a temperature. This leads the model to learn representations that maximise the similarity of matching pairs and minimise it for non-matching ones. We give a formal statement of this objective in Section 3.2, where we also use the same loss function to pre-train our connector. Figure 2.2 illustrates the objective.

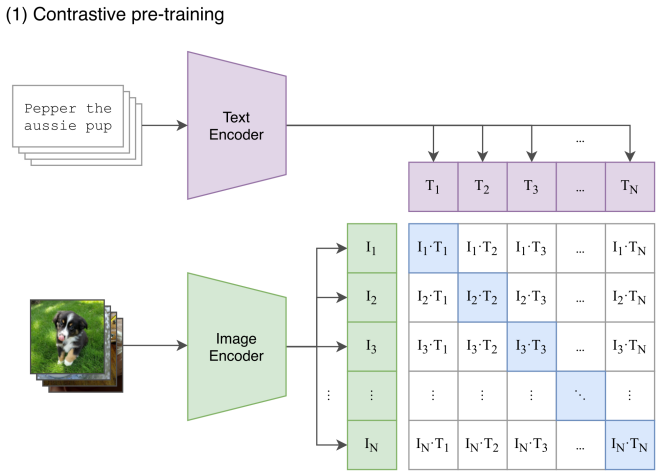


Figure 2.2: Contrastive pre-training in CLIP. Within a batch of N pairs, the matched image-caption products on the diagonal are the positives and every off-diagonal product is a negative. Source: [3].

Two immediate implications of this setup are worth noting before proceeding. First, embeddings get ℓ_2 normalised before comparison. This means that the representation occupies the unit hypersphere and only the vector direction is meaningful information. Second, the temperature is not a neutral rescaling: it dictates the strength with which non-matching pairs repel each other, the quantity Wang and Isola [16] call uniformity, and will come back in Section 2.2 as one of the quantities that determine the geometry of the learned space.

2.1.3 Multimodal large language models

A MLLM takes a pre-trained language decoder and equips it with vision capabilities, such that instead of producing similarity scores, the output of the model is text. As we have seen in recent literature surveys [14], such a model typically consists of three parts: a frozen modality encoder, a language backbone, and an interface between them that maps encoder outputs into the backbone's input space.

The training procedure that LLaVA [1][9] started with, and which most subsequent works reuse, happens in two stages. First, only the connector is trained such that its output ends up somewhere the frozen decoder can read it. The connector and the language backbone are then tuned together on instruction data under a next-token prediction loss, which means the only signal reaching the connector is backpropagated from the text it helps produce. BLIP-2 [2] splits up training similarly while implementing a different interface module, and Bunny series [17] runs the same 2-stage recipe with a smaller dataset. The backbone chosen for this thesis is LLaMA-2-7B [18] fine-tuned with LoRA [19] attachments over its quantised base weights [20]. The protocol followed is described in Section 3.2. Figure 2.3 shows this architecture.

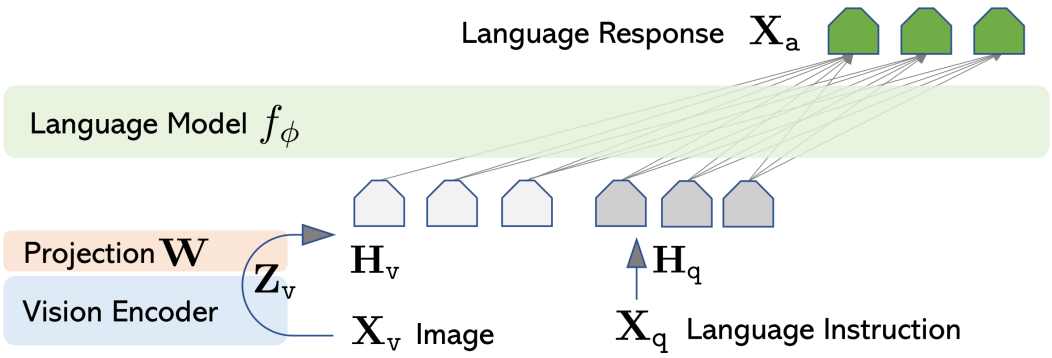


Figure 2.3: The LLaVA architecture. A frozen vision encoder maps the image $\mathbf{X}_v$ to features $\mathbf{Z}_v$, a single projection $\mathbf{W}$ maps those to $\mathbf{H}_v$ in the same input embedding space that the instruction $\mathbf{X}_q$ occupies as $\mathbf{H}_q$, and the language model f_ϕ attends over the concatenated sequence to emit the response $\mathbf{X}_a$. Source: [1].

A desirable property we will carry through to our models is that in this family of models, the decoder's own weights are frozen, and it reads the output of the connector directly. Whatever geometry it produces must be geometry that the decoder can read. That geometry can be measured directly from the projected representations, but unlike a dual encoder this family provides no similarity score against which it can be judged, so its quality is visible only in the text the model generates.

2.2 Geometric properties of the embedding space

The geometry of this shared embedding space is not arbitrary in how these models function, and one aspect has been studied in particular: the systematic gap between where images are and where texts are. We review this phenomenon and the explanation offered to account for it, and then revisit the solutions proposed to reduce it. A formal definition of the statistics used to quantify this geometry in the current work can be found in Section 3.3.

2.2.1 The modality gap

Liang et al. [4] found empirically that the two modalities of a contrastive model do not actually share the region of space that they are training to share. Image embeddings cluster apart from caption embeddings; authors report this effect in models trained on natural images and video, on medical images, and on amino-acid sequences embedded with a separately trained captioning model. They define the *gap* as the difference between the mean of ℓ_2-normalised image embeddings and mean of ℓ_2-normalised caption embeddings, and measure its size as the Euclidean distance between these two means. Figure 2.4 reproduces the evidence they report.

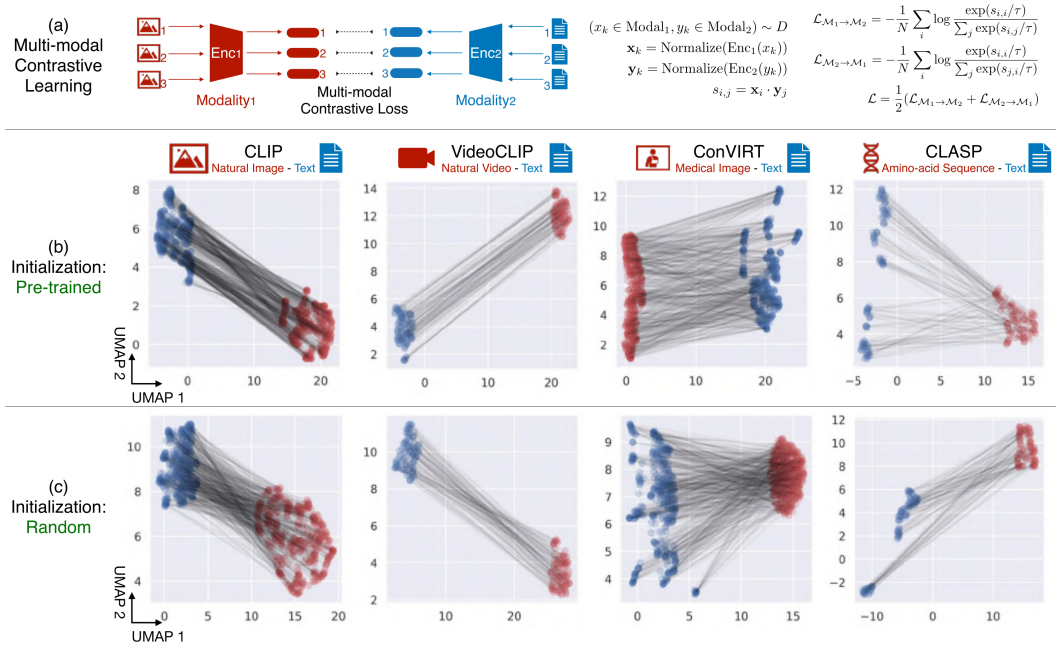


Figure 2.4: The modality gap in multimodal contrastive learning. (a) Two encoders map each modality onto the hypersphere under a symmetric contrastive loss. (b) UMAP of paired embeddings from pre-trained models, where lines join matched pairs: the two modalities occupy disjoint regions across natural images, video, medical images and amino-acid sequences. (c) The same architectures with random weights, showing the gap is present before any training. Source: [4].

The explanation offered is threefold:

- **The cone effect.** Embeddings extracted from the last layer of a network of substantial depth will generally exist within a narrow cone. This forces the average cosine similarity between random pairs to be dramatically higher than zero. Interestingly, the property persists for trained networks, even randomly initialized networks, and even random noise fed into the input layers, where it vanishes if a nonlinear activation is not used. Deep networks progressively narrow this cone at each successive layer, a phenomenon that depends strictly on the presence of nonlinear activation functions.

- **Initialisation.** The cones produced by different random initializations are different. Since each encoder in a two-tower model is initialized independently, image and text embeddings already live in separate cones before training has even started.

- **The contrastive objective.** Training preserves that separation rather than removing it. Visualizing similarity on 5000 MSCOCO [21] pairs, taking an image-text pair and shifting both embeddings towards each other before normalizing to unit length *increases* the contrastive loss, and separation achieved at training's end (distance of 0.82 at CLIP's learned temperature of $\tau = 1/100$) corresponds exactly to the minimum of that loss. Higher temperatures cause the structure to vanish. Figure 2.5 reproduces the loss curves behind this last point.

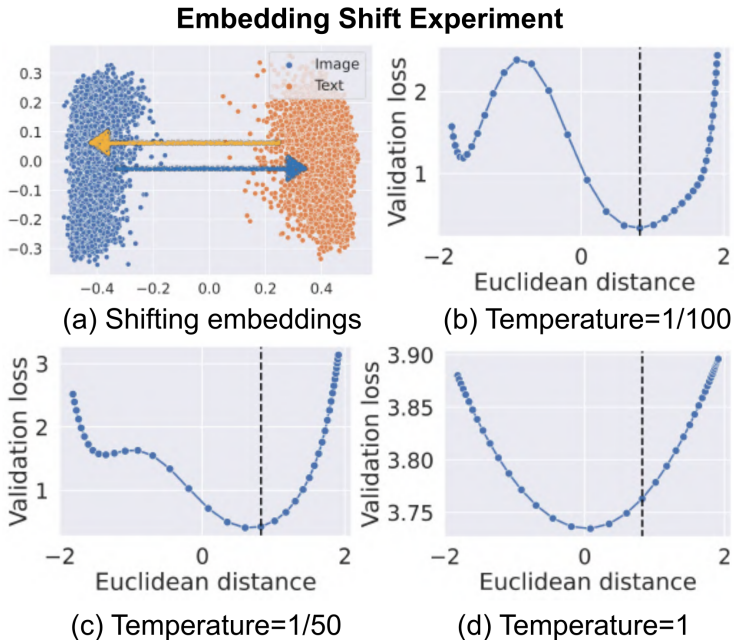


Figure 2.5: Contrastive learning preserves the gap. (a) Both modalities are shifted towards each other and renormalised. (b) At CLIP's learned temperature $\tau = 1/100$ the validation loss is minimised at the separation the model actually settles at, marked by the dashed line, so closing the gap makes the loss worse. (c) and (d) The minimum moves away from that separation as the temperature rises, and the structure disappears at $\tau = 1$. Source: [4].

A further experiment separates the contributions of the last two, initialisation and the objective [4]. When the gap is removed at initialisation by an orthogonal Procrustes map, so that the initial separation is 0.039, training grows it back to 0.746, against 1.299 for the untouched model. Initialisation and optimisation therefore each account for part of the final value.

Liang et al. also report that modifying the gap changes downstream behaviour: shifting the embeddings improves zero-shot accuracy on several classification tasks, and increasing the separation reduces denigration harms in a fairness audit at a negligible cost in accuracy. We note that their stated goal is to understand the phenomenon rather than to propose a method for closing it, and that every consequence they measure is read on a task which is itself an inner product in the same space.

2.2.2 Strategies for reducing the gap

The explanation is not the only one on offer, and the disagreement is about the cause rather than the phenomenon. Shi et al. [22] attribute the separation to insufficient training rather than to initialisation. Schrodi et al. [23] trace it instead to an information imbalance between the paired modalities, and derive object bias from the same source. Fahim et al. [24] go further and dispute the name: they report a comparable separation between two encoders of the *same* modality, conclude that the contrastive objective produces it regardless of what is being encoded, and propose

calling it a contrastive gap arising from low uniformity in the sense of Wang and Isola [16]. The present work takes no position on the cause. It measures the geometry a trained connector produces and asks which part of it a decoder is sensitive to, a question that is unaffected by why the separation arose.

The concern is also older than the multimodal literature. Representations produced by likelihood-trained language models were already known to collapse into a narrow, anisotropic cone, a phenomenon named the representation degeneration problem by Gao et al. [25] and measured across contextual encoders by Ethayarajh [26]. Mu and Viswanath [27] showed that simply removing the mean and a few dominant directions improves such representations, which is a unimodal precursor of the centroid and subspace corrections reviewed below. The statistics used in Chapter 3 to describe the image cloud, its total variance, how concentrated that variance is, and where its leading directions point, are the quantities that literature introduced.

A subsequent line of work treats the separation as something to be corrected before the representation is used, and it has moved through four positions.

- **Centroid correction with isotropic noise.** The method C3 from Zhang et al. [5] models the gap as a centroid shift plus isotropic noise sampled at random, and corrects for it by translating one modality to the centroid of the other, then adding Gaussian noise.

- **Anisotropic correction of the first two moments.** Yu et al. [6] refer to the above as the isotropic fallacy. They construct a reference frame from the leading eigendirections of the summed probe covariance matrix. By splitting the space into the dominant subspace and its orthogonal complement, then decomposing the gap vector into components within each, they are able to separately analyze a first-order mean bias vector and a second-order residual within each subspace. They observe that although centroid differences almost disappear at convergence, the residual *retains* rank in its spectrum throughout training rather than converging to white noise. From these empirical findings, they are able to construct ReAlign: a simple, training-free transformation that operates on frozen embeddings in three steps: a centering of text embeddings followed by translation to the image centroid; a rescaling of residuals by the single scalar

  $$s \;=\; \sqrt{\frac{\operatorname{tr}\Sigma_{\text{img}}}{\operatorname{tr}\Sigma_{\text{txt}}}},$$

  which equalizes total variance while maintaining spectral shape; and finally undoing the mean shift reintroduced by renormalization onto the sphere through a centroid correction. They apply the transformation only on the text side, leaving the image embeddings as an anchor.

- **Correction along dominant residual directions.** Yu et al. [7] argue that only the residual is important, and perform an incomplete correction along just the dominant residual directions.

- **Retraining the encoders.** Eslami and de Melo [28] do not post-process frozen embeddings at all. They change the pre-training objective, sharing parameters between the two towers and adding an intra-modality separation term, and retrain CLIP from scratch. This closes the gap at its source but forgoes the main attraction of the other three, which is that they cost no training.

Figure 2.6 shows the ReAlign pipeline.

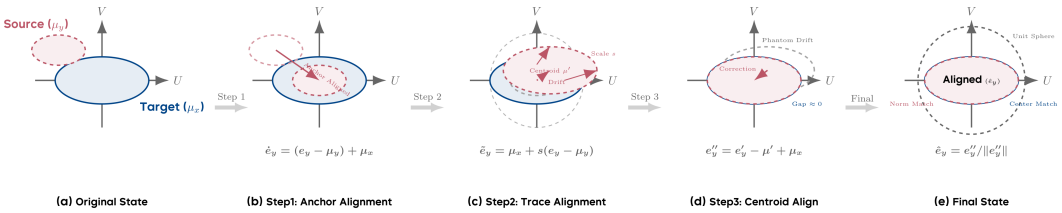


Figure 2.6: The ReAlign pipeline. Anchor Alignment translates the source modality onto the target centroid, Trace Alignment matches total residual energy by a single scalar, and Centroid Alignment removes the drift that spherical renormalisation reintroduces. Source: [6].

The second and third of these aim at substitution. Projected into the image distribution, text embeddings can be used as surrogates for images. This enables training a MLLM from unpaired data and furthers the text-only approach of Unicorn [29].

The diagnostic evidence used to arrive at the third position warrants further discussion, as it leads to an ordering opposite to the reported one. Measured across 10^4 paired embeddings, Yu et al. [7] note that image and text encodings already agree about their main geometry prior to projection, as the spectra of their covariances decay at a similar rate (correlation of 0.845) and their leading subspaces mostly align far from what would be expected by chance (overlap score of 0.441 at $q = 128$ against a null benchmark of 0.100). Upon projecting text embeddings onto the image centroid, they show that 89% of the distance remains, that the residual is strongly anisotropic with a maximum eigenvalue 28.6 times larger than the mean eigenvalue and that it occupies a subspace of effectively dimension $0.284d$. Consequently, they infer that the gap isn't primarily explained by a centroid effect, but rather by a directional variance residing in a constrained subspace. Figure 2.7 collects that evidence.

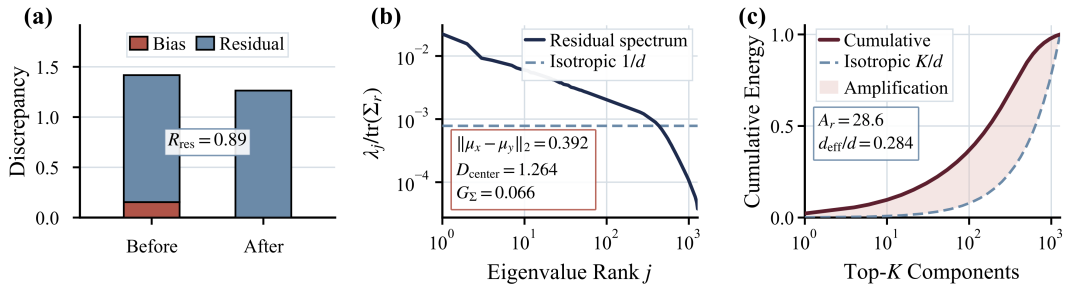


Figure 2.7: Evidence for an anisotropic residual. (a) Translating the text embeddings onto the image centroid leaves 89% of the discrepancy in place. (b) The residual spectrum departs from the isotropic baseline $1/d$. (c) Residual energy concentrates in a low-dimensional subspace, with $A_r = 28.6$ and $d_{\text{eff}}/d = 0.284$. Source: [7].

Two caveats adjust that statement. The first is about the object under decomposition. Note that in both papers the gap vector is computed per matched pair, as the difference between the embeddings of the same sample. The residual therefore measures how far apart their two encoders place a corresponding image and caption. This kind of residual cannot be undone by any method operating at the distribution level, as a single overarching transformation can't align one data point with its match, and the discrepancy would persist even when comparing two datasets with the exact same statistical properties. The quantity optimized in this thesis is distributional, on the other hand: the centroid distance between the two clouds and each cloud's covariance when treated individually. The two statements are about different objects and thus not contradictory. The second caveat is about the criterion used. Both methods are tested on whether a transformed text vector can substitute for an image, as measured by average performance over general, reasoning, and hallucination benchmarks. The present work keeps real images at test time and instead asks if reshaping the image representation aids a decoder whose base weights are frozen in describing them.

To conclude, the works reviewed in this subsection share a common shape and a common limitation. Each characterises the gap as a geometric object with more than one component, and each corrects several of those components at once through a fixed map estimated over a corpus of frozen vectors, reporting a single aggregate outcome. As a consequence, none of them divides the credit between the components it moves, none can say what happens when a trainable connector is asked to produce a given geometry under a language objective competing for the same parameters, and none reshapes the image representation itself, the transformed text embedding standing in for the image wherever the correction has been carried into a generative model [6][7]. The experimental design presented in Chapter 3 is intended to address these three points together.

2.3 Image captioning

Image captioning prompts the model to generate a free-form description of an image, a task posed in its modern form by Vinyals et al. [30]. Captioning is the task used to evaluate every condition in this thesis. As the output is not numeric but a sentence, its quality must be approximated and no current measure is considered sufficient. We provide background on metric families used, and then describe the specific measure of object hallucination that we report with them. The exact definitions and evaluation protocol we use can be found in Section 4.2.

2.3.1 Evaluation metrics

Three families of caption quality metrics used, which pairwise correlate with each other less than would be expected from their shared joint application:

- **N-gram overlap.** Reference-based metrics in this category score a candidate caption against a set of human-produced references based on surface similarity. BLEU [31] counts matching n-grams; METEOR [32] extends this to stemmed and synonymous n-grams; CIDEr [33] up- and down-weights each n-gram based on how frequently it appears across the corpus of reference captions.
- **Semantic matching.** Also reference-based, SPICE [34] scores candidate captions by parsing both them and the references into scene graphs of objects, attributes and relations, then comparing graph tuples instead of surface strings.
- **Reference-free scoring.** CLIPScore [35] scores candidate captions by matching directly against the image through a contrastive model, requiring no reference caption.

The third family requires care in a study of the modality gap. Scoring with the same encoder that a model was trained against would be circular, since the score would reward the very geometry under manipulation. A different backbone is therefore used for this purpose, as reported in Section 4.2.

2.3.2 Object hallucination

None of these metrics distinguish whether a caption describes objects that aren't actually depicted. This failure mode has received significant attention recently as generative multimodal models reach a degree of fluency where their errors are superficially plausible. Rohrbach et al. [36] proposed CHAIR as a means to measure it directly on MSCOCO data [21]. The metric extracts mentions of objects from a caption, maps them via a synonym list to object categories present in the dataset, then counts those not mentioned by the image annotation. It is reported at two levels of granularity: the fraction of mentions that were wrong, and the fraction of captions that contain at least one wrong mention.

Its usage is subject to one caveat that applies to all hallucination rates of this variety. By mentioning fewer objects, a model can artificially reduce its score. As such, the rate is only meaningful when compared to some measure of caption verbosity. Both are reported throughout this thesis. A complementary line of work avoids the problem by not scoring free text at all: POPE [37] instead polls the model with balanced yes/no questions about whether a given object is present, which removes the dependence on generation length and style. It is discriminative rather than generative, and is therefore not a substitute for a captioning measure; Section 4.2 explains why this thesis reports CHAIR.

Chapter 3

Methodology

The question motivating this chapter is whether minimizing the modality gap at the input to the language model helps image captioning, and if so, which aspect of the gap it is. In order to answer this question, we need to be able to vary one geometric quantity at a time while leaving the others unchanged, which means we need to train the module outputting the visual representation from a known initialization and measured at every step.

That is why we develop our own multimodal large language model rather than fine-tuning an existing one. A released checkpoint leaves no control over the geometry of the connector output layer: its axes have been decided, are fixed, can no longer be pinned during training, and there is no corresponding untreated reference computed with the exact same recipe against which a change could be read. Developing our own connector from scratch gives us a starting point with three axes pinned to setpoints and orientation held by the absence of a contrastive term, plus variations that differ in exactly one axis. The trade-off is scale, as discussed in Section 4.1.

Training proceeds through two stages, identical to those of LLaVA and discussed in Section 3.2. In Stage 1, only the connector is trained using a contrastive objective that aligns the pooled visual representation with the embedding of the caption it is paired with. In Stage 2, the connector is trained together with the low-rank adapters in the language model using an autoregressive objective for response tokens. The modality gap is measured at the connector's output, where the two stages are connected and the only point where the two modalities are commensurable.

This chapter introduces the architecture, the training protocol in two stages, how the modality gap is measured on this architecture, the objectives that push each component of this gap, and the best configuration. The experimental protocol and ablation studies supporting each component are provided in Chapter 4.

3.1 Architecture

All experiments use a connector-based vision-language model composed of a frozen visual encoder, a trainable connector, and an LLM whose weights are frozen apart from trained low-rank adapters [19] (Figure 3.1). This architecture is minimal, with

the connector representing the only trainable link between modalities, guaranteeing that any modification to the geometry of the visual representation passes through this module, enabling the ablation study.

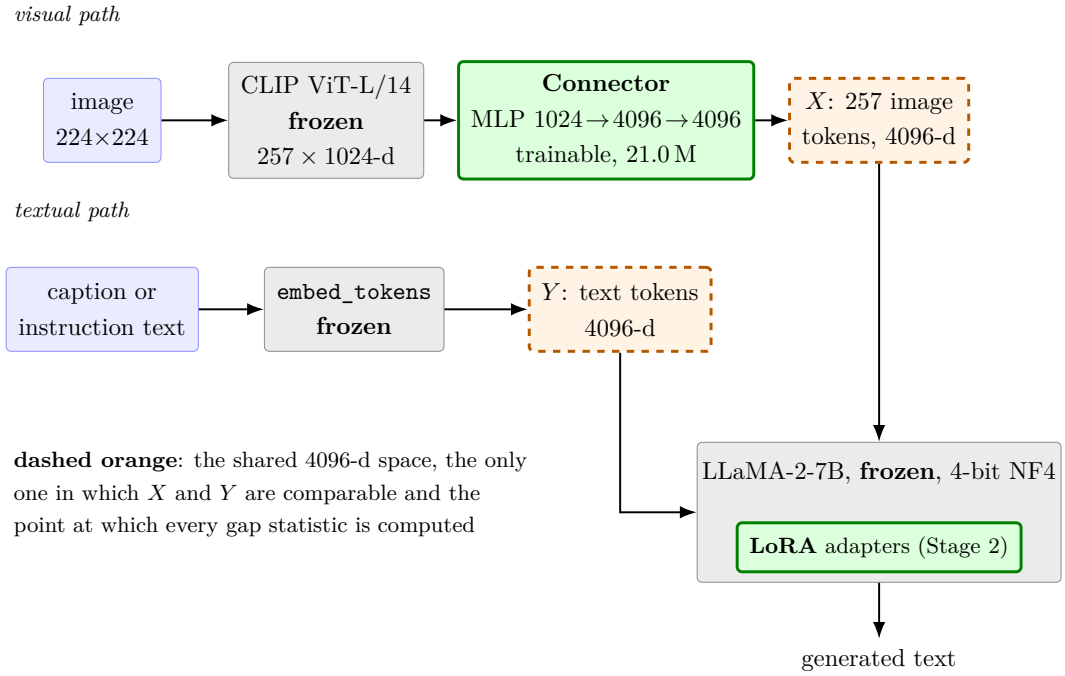


Figure 3.1: The model architecture. The connector is the only module that is trained in both stages, and its output is an input to the LLM and is where the gap was measured. All the gap metrics were computed in float64.

Table 3.1 gives the configuration of each module.

Component	Configuration
Visual encoder	`openai/clip-vit-large-patch14` [3], 224×224, frozen
Visual tokens	257 = 1 `[CLS]` plus 16×16 patches [38], 1024-d
Connector	2-layer MLP with GELU [39], 1024 → 4096 → 4096, 21.0 M parameters
Language model	`meta-llama/Llama-2-7b-hf` [18], a decoder-only transformer [40], hidden size 4096, frozen
Quantisation	4-bit NF4 with double quantisation [20], bfloat16 compute
Adapters	LoRA r=8, α=16, dropout 0.05, on `q_proj`, `v_proj`, `o_proj`

Table 3.1: Model configuration.

3.2 Training protocol

Training was built in two stages (Figure 3.2, Table 3.2); the optimiser settings that implement them are experimental detail and are reported in Table 4.1.

Stage 1: alignment. Only the connector is trained. Each image is encoded by the frozen visual tower, projected, and reduced to a single vector by taking its `[CLS]` token; the paired caption is passed through the frozen embedding table and mean-pooled. A symmetric InfoNCE objective [15] pulls matched pairs together and pushes the rest of the batch apart. Nothing else moves, so this stage sets where the visual cloud sits relative to a text side that is held still. It is also what makes orientation a manipulable axis: without a contrastive stage there is no aligned reference to rotate towards.

Stage 2: instruction tuning. The connector is refined together with low-rank adapters in the language model [19], under an autoregressive cross-entropy loss computed on response tokens only. The visual tokens enter as ordinary input embeddings, so on its own this stage gives the connector no direct signal about the geometry it produces: its gradients arrive exclusively through the effect of those tokens on the text the model generates. The geometry terms of Section 3.5 are added at this stage, and supply that signal directly.

Relation to existing recipes. The two-stage setup follows LLaVA [1][9], whose first stage trains only the connector under an autoregressive loss and whose second stage trains the connector and the LLM together. BLIP-2 [2] uses the same split, but its first stage optimises three objectives jointly, only one of them contrastive, and against a trainable text transformer inside the Q-Former. Our first stage keeps the contrastive part alone: a symmetric InfoNCE objective [15] against the language model's frozen embedding lookup table, which is CLIP [3] with the text tower replaced by a fixed one. That choice is what makes orientation a manipulable axis, since the orientation intervention of Section 3.5 reuses the same objective as its drive. The data and schedule follow previous work on reducing the gap [6][7], which corrects frozen embeddings with a fixed transformation rather than with a training term.

	Stage 1	Stage 2
Objective	symmetric InfoNCE	AR cross-entropy
Loss positions	`[CLS]` and pooled text	response tokens only
Trainable	connector	connector and LoRA

Table 3.2: The two stages side by side. They differ in objective, in where the loss is computed, and in what is allowed to move. The optimiser settings that implement them are experimental detail and are reported in Table 4.1.

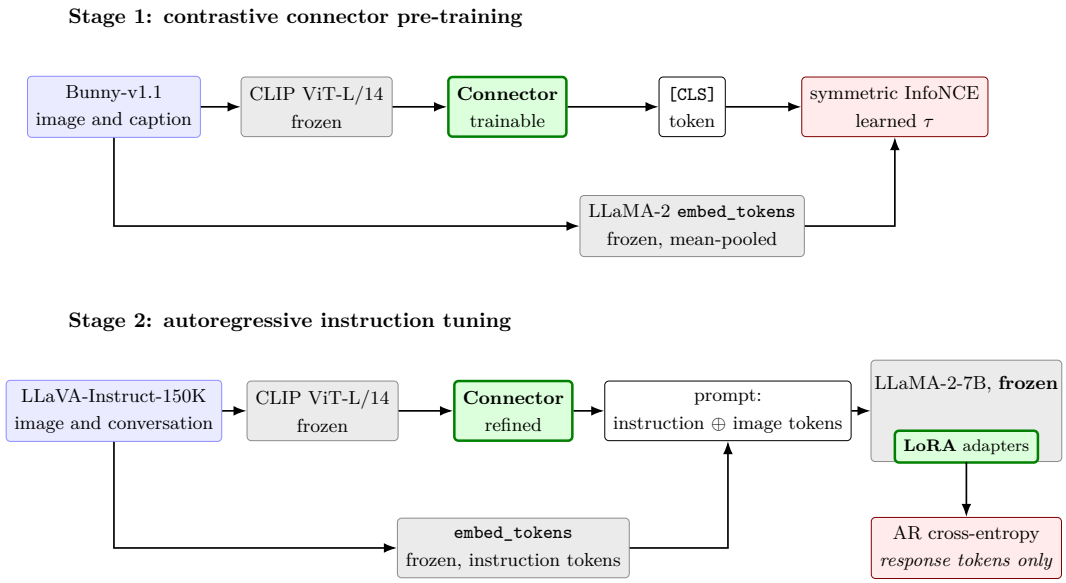


Figure 3.2: The two training stages. Stage 1 uses only the LLM's embedding lookup table. Stage 2 uses a loss computed only on the response tokens, and the connector receives gradients exclusively via its effect on the generated text.

3.3 Measuring the gap

The question is whether reducing the gap at the input of the language model improves image captioning. That fixes where the gap has to be measured: at the connector output, because that is the input the decoder actually receives. It is also the only place the two modalities are commensurable, since the visual tokens have been projected into the decoder's embedding space and the caption tokens are read from the decoder's own embedding table. A gap measured earlier, inside CLIP's own space for instance, is a gap the decoder never sees.

We therefore measure at the connector output. The image representation is the result of averaging the 257 projected visual tokens, while the text representation comes from averaging the embedding-table lookups over the caption's content tokens. This provides two different clouds in space: $X, Y \in \mathbb{R}^{N\times 4096}$.

3.3.1 The proposed four axes

Let $\bar{x}, \bar{y}$ be the cloud means and Σ_X, Σ_Y the covariances, with $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$ the eigenvalues of Σ_X and $d = 4096$.

Location The centroid distance, the quantity written as "the modality gap":

$$G_\mu := \|\bar{x} - \bar{y}\|, \qquad \widehat{G}_\mu := \frac{G_\mu}{\sqrt{\operatorname{tr}\Sigma_X}}. \tag{3.1}$$

Rescaling both clouds by c multiplies numerator and denominator alike, so $\widehat{G}_\mu$ is scale-free and is used whenever two representation spaces are compared. Within one space the raw G_μ is quoted, since it is what the objective acts on.

Scale The total variance $s := \operatorname{tr}\Sigma_X = \sum_j \lambda_j$, the spread of the cloud summed over all d directions. Scaling Σ_X to $c\Sigma_X$ multiplies s by c, so it has no scale-free form of its own; the objective therefore holds it as a ratio to its value at initialisation rather than against an absolute target.

Shape The effective rank [41], taken throughout as the participation ratio

$$r := \frac{(\operatorname{tr}\Sigma_X)^2}{\operatorname{tr}(\Sigma_X^2)} = \frac{(\sum_j \lambda_j)^2}{\sum_j \lambda_j^2} \in [1, d], \tag{3.2}$$

which equals 1 if all the variance falls on one line and d if the cloud is isotropic, and thus may be interpreted as the number of effective directions of the variance. Both numerator and denominator are homogeneous of degree two in Σ_X, and therefore scaling Σ_X to $c\Sigma_X$ scales them both by c^2 but leaves r unaffected, despite the fact that the eigenvalues themselves are scaled to $c\lambda_j$. Scaling and shape are thus separate coordinates: the trace changes with r constant under scaling, and r changes with trace constant under shifting the eigenvalues. Keeping both constant thus requires two penalties, which is why the baseline has one of each.

Orientation With $U_X^{(q)}, U_Y^{(q)}$ the leading q eigenvectors,

$$O_q := \frac{1}{q}\|U_X^{(q)\top} U_Y^{(q)}\|_F^2 = \frac{1}{q}\operatorname{tr}(P_X P_Y) \in [0, 1]. \tag{3.3}$$

The singular values of $U_X^{(q)\top} U_Y^{(q)}$ are cosines of the principal angles between the subspaces. Raw values grow with q for purely dimensional reasons. For two independently and uniformly drawn q-dimensional subspaces $\mathbb{E}[P_Y] = (q/d)I$ by rotational symmetry, so

$$\mathbb{E}[O_q] = \tfrac{1}{q}\operatorname{tr}(P_X \tfrac{q}{d} I) = q/d. \tag{3.4}$$

Overlap is thus interpreted as a multiple of chance, the results using $q = 16$, where chance is $16/4096 \approx 3.9 \times 10^{-3}$.

3.3.2 Reduction of the diagnostic suite

The diagnostic computes twenty scalars per checkpoint, not four. They are listed along with the range taken in Table 3.3 for each of the eight conditions trained. The four axes of Section 3.3 are the survivors of three constraints imposed on this suite.

Non-degeneracy. An axis must take different values across the eight conditions, or it cannot account for anything. This excludes the covariance-shape discrepancy G_Σ, which reads 1.000 in all eight, and the k-nearest-neighbour mixing rate, which is exactly zero in seven of them.

Non-redundancy. An axis must not be an algebraic function of one already retained, or the factorial reports the same effect twice. This excludes the bias

centroid γ, whose norm is G_μ by construction, and the effective dimension d_{eff}, which tracks the participation ratio to within 0.07%.

Controllability. An axis must be drivable or holdable by a differentiable penalty on a single training batch, which for the geometry leg is 64 images. This excludes the condition number κ: sixty-four vectors in $d = 4096$ give a covariance of rank at most 63, so κ is undefined on every batch the objective sees, not merely noisy.

The proposed four axes are also complete for this model. The text side is frozen, so the gap depends only on the mean difference $\bar{x} - \bar{y}$ and the image covariance Σ_X. The mean difference gives the centroid offset; eigendecomposing $\Sigma_X = U\Lambda U^\top$ gives the total of Λ, its concentration, and where U points. No fifth quantity is available without leaving that pair. Completeness is not exhaustiveness, and the shortfall is worth stating: the trace and the participation ratio are two summaries of a d-dimensional spectrum and do not determine it, so residual spectral structure is left uncontrolled throughout.

The other sixteen fail one of the three requirements. Three are saturated: beyond G_Σ and the mixing rate already named, C_λ spans only 0.016. The saturation of G_Σ has a simple cause, in that $\operatorname{tr}\Sigma_Y = 0.0519$ against 252 to 4790 on the image side collapses the ratio to one. Five are already determined by axes kept: besides $\|\gamma\|$ and d_{eff}, $\|\beta\|^2 = G_\mu^2 + \varepsilon$, the pair residual energy ε tracks $\operatorname{tr}\Sigma_X$ to within 0.09%, and the residual ratio is a function of two of them. The remainder are either constants of the frozen text side or measurable on a corpus but not on a batch: the angular Jensen–Shannon divergence needs some 2×10^4 pairs where a batch offers 2016, and A_r weights the leading eigenvalue alone, which is not differentiable where that eigenvalue is degenerate.

The four kept are cheap on exactly the batch the geometry leg sees. The centroid is a mean, the trace a sum of squared deviations, and the participation ratio comes from the 64×64 Gram matrix of the centred batch, whose non-zero eigenvalues are the covariance's up to the constant factor the estimator divides by, and that factor cancels in the ratio. A rank-16 subspace is well determined by the 63 directions a batch of this size provides.

Quantity	Range over the eight conditions
Retained as axes	
Centroid distance G_μ	30.97 to 245.46
Image trace $\operatorname{tr}\Sigma_X$	252.7 to 4789.6
Image participation ratio r_X	20.21 to 39.19
Subspace overlap O_{16}	0.0041 to 0.1114
Constant: properties of the frozen text side	
Text trace $\operatorname{tr}\Sigma_Y$	0.05188
Text participation ratio r_Y	179.19
Text exponent α_Y	1.5752
Text condition number κ_Y	1.56×10^9
Degenerate: no range across conditions	
Covariance discrepancy G_Σ	1.00000
k-NN mixing rate, k=20	0.000 to 0.010
Spectral correlation C_λ	0.871 to 0.887
Redundant: determined by a retained axis	
Bias centroid $\|\gamma\|$	equals G_μ
Bias magnitude $\|\beta\|$	75.80 to 254.62
Pair residual energy ε	252.7 to 4786.8
Residual effective dim. d_{eff}	20.21 to 39.20
Residual ratio	0.028 to 0.833
Not controllable on a single batch	
Image exponent α_X	1.790 to 2.341
Image condition number κ_X	1.3 to 160×10^{12}
Angular JS divergence	0.102 to 0.205
Residual anisotropy A_r	276.1 to 632.1

Table 3.3: The complete diagnostic suite, measured on MSCOCO `val2017`, n=5000, d=4096, over the eight conditions `C1_stage2`, `C2_stage1`, `C3_stage2`, `C3pinr`, `Cloc`, `Corient`, `Crank15` and `Cscale1500`. Only the first block satisfies all three requirements and defines the factorial design; the reason for each exclusion is given in the text.

3.4 Control must act on the measured object

Any geometric objective needs to condense the 257-token sequence into one vector. There are two main ways to do this: using the `[CLS]` token, which acts as the standard CLIP summary and is used by the Stage-1 objective, or taking the mean of all 257 tokens. The metrics use the mean, which is calculated as `proj_tokens.mean(dim=1)`. Meanwhile, the decoder processes the whole sequence. A loss that focuses on `[CLS]` is therefore controlling something different from what is actually measured and read. Focusing on `[CLS]` does not guarantee any effect on a pooled measurements that is mostly influenced by the 256 patch tokens. As a result, every condition below applies its geometry terms to the pooled-257 vector. Section 4.5.4 shows what mismatch costs.

3.5 Geometry objectives

The constraints introduce geometrical expressions into the Stage-2 goal. Here, $u_{b,t}$ refers to the projected token t of batch example b, $\bar{z}$ is the batch average of the vectorized pooled examples, and s and r denote the batch trace and participation ratio, matching the definitions of Section 3.3.

Location drive A quadratic pull of the batch centroid towards the frozen text centroid μ_y, with τ fixed at the post-Stage-1 image trace:

$$\mathcal{L}_{\text{dist}} := \frac{\|\bar{z} - \mu_y\|^2}{\tau}, \qquad \frac{\partial \mathcal{L}_{\text{dist}}}{\partial u_{b,t}} = \frac{2}{\tau BT} (\bar{z} - \mu_y). \tag{3.5}$$

The gradient is constant across all tokens and independent of b or t. All tokens undergo the same transformation; thus, the requested update is a simple rigid translation. As translation does not alter the covariance, this term influences only the centroid. This allows us to consider location separately. The essential point is that this update acts on the parameters of a nonlinear connector, not directly on the token cloud. Thus, we must empirically confirm that covariance remains unchanged, and we report these values for each condition. The freezing of τ is crucial. If we recalculated it for each batch, the optimization process could trivially lower the loss simply by increasing the cloud's spread.

Pins The same structure applies, but here the target is replaced by the frozen value. The location pin holds the batch centroid at μ_{x_0}, its post-Stage-1 value, while scale and shape are held relative to their own frozen setpoints:

$$\mathcal{L}_{\text{pin}} := \frac{\|\bar{z} - \mu_{x_0}\|^2}{\tau}, \qquad \mathcal{L}_{\text{scale}} := \left(\frac{s}{s_0} - 1\right)^2, \qquad \mathcal{L}_{\text{rank}} := \left(\frac{r}{r_0} - 1\right)^2. \tag{3.6}$$

Each term's minimum occurs only at its respective setpoint, and it penalizes any deviation. It is essential to use a relative form due to the two orders of magnitude difference between s and r, which stand at 4767.3 and 23.31 at the setpoints. Thus, a single weight may be used for both. Once the setpoint is moved, the pin becomes a drive. This method generates Cscale1500 and Crank15.

Orientation drive A contrastive term $\mathcal{L}_{\text{NCE}}$ applies the same objective as Stage 1, but to the pooled vector rather than to `[CLS]`. It measures the cosine similarity of normalized vectors. This method does not depend on scale and focuses on the angular relationship between the two clouds.

The combined loss function The driven geometry term enters through a convex combination, so that the optimization budget is conserved and the term is paid for out of the language objective rather than added on top of it:

$$\mathcal{L}(\theta) = (1 - \lambda)\, \mathcal{L}_{\text{AR}}(\theta) + \lambda\, \mathcal{L}_{\text{geo}}(\theta), \qquad \lambda \in [0, 1]. \tag{3.7}$$

The pins are not part of that trade. They carry a fixed weight of 1.0 and sit outside the combination, as Equation (3.9) shows, since their role is to hold an axis in place rather than to compete with the language objective for budget. Here, λ is a fixed hyperparameter for each condition and is set from outside the model. It cannot be learned usefully. If we try to make it trainable by setting $\lambda = \sigma(\psi)$ and then apply the chain rule, we get:

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \mathcal{L}_{\text{geo}} - \mathcal{L}_{\text{AR}}, \qquad \frac{d\lambda}{d\psi} = \lambda(1-\lambda), \qquad \frac{\partial \mathcal{L}}{\partial \psi} = (\mathcal{L}_{\text{geo}} - \mathcal{L}_{\text{AR}})\,\lambda(1-\lambda). \tag{3.8}$$

Because $\mathcal{L}_{\text{geo}}$ stays numerically far below $\mathcal{L}_{\text{AR}}$ throughout training, 0.06 against 1.19 at the end of the longest run, the bracket is negative and gradient descent drives λ towards 1, abandoning the language objective entirely. The weight collapses towards whichever term is smaller on its own scale, which says nothing about which term matters.

Figure 3.3 shows what each of the four interventions does to the image cloud.

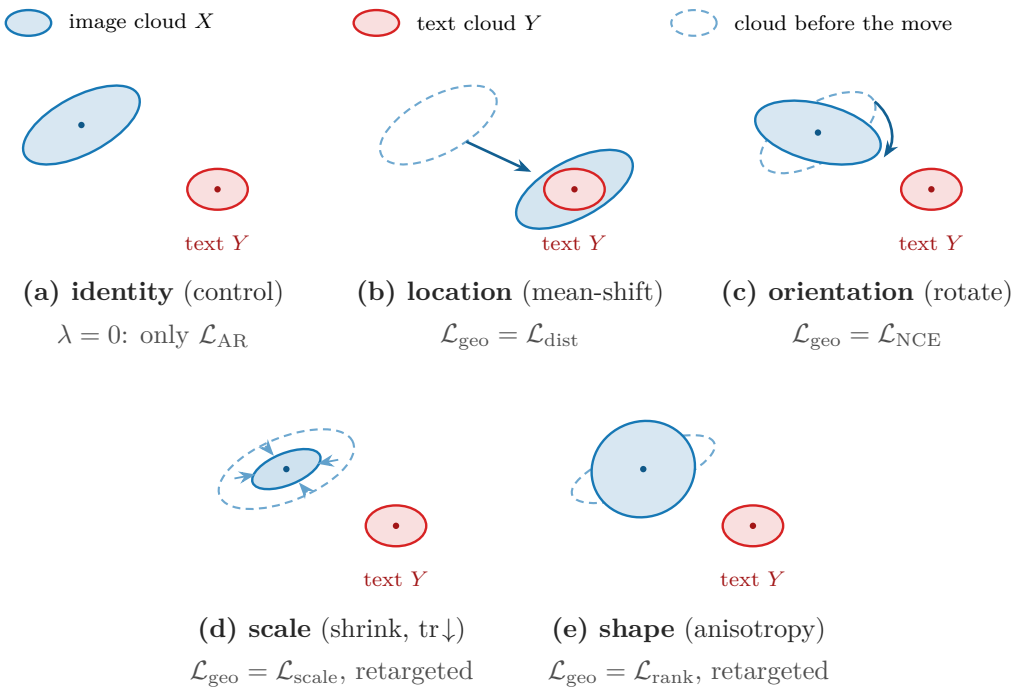


Figure 3.3: The five training conditions as geometric interventions on the image cloud. In each panel the dashed ellipse is the cloud before the intervention and the solid one after it, while the text cloud Y is fixed throughout because it comes from the frozen embedding table. Every condition is Equation (3.7) with the $\mathcal{L}_{\text{geo}}$ named beneath its panel, and the three axes it does not drive are held at their post-Stage-1 setpoints by the pins of Equation (3.6). Panel (a) is the baseline: all four axes are pinned, only the language objective is free, and it therefore carries no dashed outline. In (d) and (e) the pin setpoint is moved rather than frozen, which is what turns a pin into a drive. The conditions are named in Table 4.5.

3.6 The proposed objective: closing the location gap

The components above combine into one recipe. Starting from the Stage-1 connector, Stage 2 runs with the autoregressive objective, a location drive towards the frozen text centroid, and pins holding scale and shape at their post-Stage-1 setpoints:

$$\mathcal{L} = 0.9\,\mathcal{L}_{\mathrm{AR}} + 0.1\,\mathcal{L}_{\mathrm{dist}} + 1.0\,\mathcal{L}_{\mathrm{scale}} + 1.0\,\mathcal{L}_{\mathrm{rank}}. \tag{3.9}$$

The drive weight comes out of the language budget, so $w_{\mathrm{AR}} = 1 - \lambda_d = 0.9$, while the two pins are additive guards that do not tax it. Every term acts on the pooled-257 vector, matching the measurement exactly. This configuration is referred to as Cloc, and Table 3.4 summarises it.

Term	Weight	Role
$\mathcal{L}_{\mathrm{AR}}$	$w_{\mathrm{AR}} = 0.9$	language modelling on response tokens
$\mathcal{L}_{\mathrm{dist}}$	$\lambda_d = 0.1$	closes the centroid distance towards μ_y
$\mathcal{L}_{\mathrm{scale}}$	$\lambda_s = 1.0$	holds total variance at s_0
$\mathcal{L}_{\mathrm{rank}}$	$\lambda_r = 1.0$	holds the participation ratio at r_0

Table 3.4: The method of Equation (3.9). One axis is driven, two are held, and all four terms act on the pooled-257 vector.

Only one axis is driven, and this choice is intentional. The reason for leaving orientation intact is the fact that a contrastive drive, in addition to the location drive, would also close the centroid even more, down from 30.97 to 11.72, making any benefit difficult to distinguish from the increased location closure. Driving only one axis makes the claim clear. Section 4.6 reports that condition and explains why it does not cleanly test what orientation brings.

There are two more decisions made in Equation (3.9) which are by no means arbitrary, and the ablations described in Chapter 4 are what justify those decisions. Neither scale nor shape is free to vary, and this is because allowing either to vary is detrimental or neutral, but never helpful. Furthermore, the drive acts on the pooled vector and not on the `[CLS]` token because, as shown in Section 3.4, applying a term to the `[CLS]` token controls a different object than the one the decoder sees.

Chapter 4

Experiments

This chapter describes the budget used for training all models, the data, and metrics, the results of the methodology described in Section 3.6, and the ablations. In an ablation, exactly one element of the methodology is either removed or modified in some way, and the loss is measured; this is why attributing an effect to a particular geometric axis is justified.

4.1 Computational budget

We processed each run sequentially on a single RTX 2080 Ti (11 GB), requiring about 20 hours per scheduled job. We restrict the second stage to 450 optimizer steps, which amounts to around 14,400 samples in total with a batch size of 32, making up roughly 9.1% of one epoch for LLaVA-Instruct-150K. The goal of the project is not to develop a competing captioning model but to understand the influence of representation space on caption quality, given our limited resources. All runs use PyTorch [42] with AdamW [43], its weight decay set to zero [43].

The training recipe for each case in this chapter is the same, except for the additional geometry term in the Stage-2 objective; therefore, the parameters defined in Table 4.1 are fixed for all cases and are not varied in the study.

	Stage 1	Stage 2
Learning rate	5×10^{-4}	2×10^{-4}
Schedule	cosine, 3% warm-up	cosine, 3% warm-up
Batch	64	4×8 accumulation (effective 32)
Precision, seed	bfloat16, 42	bfloat16, 42

Table 4.1: Optimiser settings. Both stages are described in Section 3.2 and compared in Table 3.2; all conditions share these settings and differ only in the geometry term added to the Stage-2 objective (Section 3.5).

4.2 Datasets and evaluation metrics

Table 4.2 lists the data used at each stage.

Dataset	Stage	Size used	Role
Bunny-v1.1 [17]	Stage 1	image/caption pairs	contrastive pre-training
LLaVA-Instruct-150K [1]	Stage 2	$\approx$ 14,400 items	instruction tuning
MSCOCO `val2017` [21]	evaluation	5,000 images	brief captioning, CHAIR
dd256 (first 1,300 images of `val2017`)	evaluation	1,300 images	detailed captioning, gap

Table 4.2: Datasets used for training and evaluation. Stage 2 includes examples from `train2017` images disjoint from the evaluation dataset `val2017`.

Evaluation runs under the two regimes of Table 4.3.

	Brief captioning	Detailed description
Dataset	MSCOCO	MSCOCO
Images	5,000	1,300 (subset)
Prompt	"Provide a brief, one-sentence description of this image."	"Describe the following image in detail."
Decoding	greedy, 64 new tokens	greedy, 256 new tokens
Used for	CLIPScore, BLEU, METEOR, CIDEr, SPICE	CLIPScore, CHAIR

Table 4.3: The two evaluation regimes. The first checks for brief responses to the provided image, while the second focuses on a detailed description decoding up to 256 tokens. The metric sets differ by design: BLEU, CIDEr, METEOR and SPICE compare against one-sentence MSCOCO references and lose their meaning at four times that length, so the detailed regime uses reference-free measures only. The gap statistics of Section 3.3 belong to neither regime, being computed from the projected embeddings of the same 5,000 images without decoding.

Three families of metrics are used, and they differ in what the generated caption is compared against: the image itself, a set of human reference captions, or the list of objects the image is annotated with.

First, *Reference-free alignment* utilizes CLIPScore [35], which is computed as $2.5 \cdot \max(\cos(v, t), 0)$ between the CLIP embeddings of the image and its generated caption using ViT-B/32 instead of the ViT-L/14 encoder (the one we already have in our architecture), to make the system non-circular.

Second, *Reference-based quality* includes SPICE, METEOR, BLEU, and CIDEr, where SPICE is the key metric because it uses scene-graph analysis rather than n-grams. Let c denote the generated caption and $S = \{s_1, \ldots, s_m\}$ denote the set of references; we define the four metrics as follows.

SPICE [34]:

$$P = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(c))|}, \qquad R = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(S))|}, \\ \mathrm{SPICE}(c, S) = \frac{2\,P\,R}{P + R}, \tag{4.1}$$

where $G(\cdot)$ is the scene graph of a caption, $T(\cdot)$ its set of tuples and $\otimes$ the tuple matching operator.

BLEU [31]:

$$\text{BLEU-}N = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \qquad \mathrm{BP} = \min\left(1,\ e^{1-\ell_r/\ell_c}\right), \tag{4.2}$$

where p_n is the modified n-gram precision, $w_n = 1/N$, ℓ_c and ℓ_r are the candidate and closest reference lengths, and $N = 4$.

METEOR [32]:

$$\mathrm{METEOR} = (1 - \mathrm{Pen})\, F_\alpha, \qquad F_\alpha = \frac{P\,R}{\alpha P + (1-\alpha)R}, \qquad \mathrm{Pen} = \gamma \left(\frac{ch}{m_u}\right)^{\beta}, \tag{4.3}$$

where P and R are unigram precision and recall, ch the number of alignment chunks, m_u the number of matched unigrams, and $\alpha = 0.9$, $\beta = 3$, $\gamma = 0.5$.

CIDEr [33]:

$$\mathrm{CIDEr}(c, S) = 10 \sum_{n=1}^{N} w_n\, \mathrm{CIDEr}_n(c, S), \\ \mathrm{CIDEr}_n(c, S) = \frac{1}{m} \sum_{j=1}^{m} \frac{\boldsymbol{g}^n(c)^\top \boldsymbol{g}^n(s_j)}{\|\boldsymbol{g}^n(c)\|\, \|\boldsymbol{g}^n(s_j)\|}, \tag{4.4}$$

where $\boldsymbol{g}^n(\cdot)$ is the vector of TF-IDF weights over n-grams.

Third, *Object hallucination* is measured with CHAIR [36], reporting CHAIR_s, CHAIR_i, and object recall with ground truth as the union of the 80 COCO-category segmentation labels in `instances_val2017` and the objects mentioned across the five reference captions.

$$\mathrm{CHAIR}_s = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{1}[|H_i| > 0], \qquad \mathrm{CHAIR}_i = \frac{\sum_i |H_i|}{\sum_i |M_i|}, \\ \mathrm{Recall} = \frac{\sum_i |\mathrm{set}(M_i) \cap G_i|}{\sum_i |G_i|}, \tag{4.5}$$

where $\mathcal{I}$ is the evaluation set, M_i is the set of object mentions in the caption for image i counted with multiplicity, G_i is the ground-truth object set, and $H_i = \{o \in M_i : o \notin G_i\}$ is the hallucination set. Differences in these three quantities are expressed using paired bootstrap intervals over images. All the metrics along with their equations and ranges are collected in Table 4.4.

Metric	Family	Eq.	Range
CLIPScore↑	reference-free	in text	$[0, 2.5]$
SPICE↑	reference-based	(4.1)	$[0, 1]$
BLEU-4↑	reference-based	(4.2)	$[0, 1]$
METEOR↑	reference-based	(4.3)	$[0, 1]$
CIDEr↑	reference-based	(4.4)	$[0, 10]$
CHAIR_s↓	hallucination	(4.5)	$[0, 1]$
CHAIR_i↓	hallucination	(4.5)	$[0, 1]$
Object recall↑	hallucination	(4.5)	$[0, 1]$

Table 4.4: The downstream metrics, their defining equations and ranges. An up arrow marks a metric where higher is better, a down arrow where lower is better.

4.3 The conditions

Each of the conditions created manipulates or modifies a part of the method in order to see what changes. Table 4.5 provides the names of the resulting conditions.

There are four blocks in the table and each of them addresses a different issue. The first is about manipulating the training protocol itself by deleting either stage. It serves the purpose of determining the origin of the geometry of the connector output before adding the geometric term. The second is about varying a contrastive alignment term which is the intervention discussed in the works from Section 2.2. It is there to demonstrate why a variation of this type does not work for the attribution. The third is factorial and it allows one axis to be released while the other three are fixed. The fourth gives the two dose-ladder rungs that appear in Table 4.7 alongside the factorial conditions.

Condition	Axis released	Objective
random	(untrained)	random connector, no training
stage2-only	(no Stage 1)	Stage 2 only, connector randomly initialised
stage1-only	(no Stage 2)	Stage 1 only
full	all free	Stage 1 and Stage 2, no geometry terms
full+orient	orientation	$(1-\lambda)\mathcal{L}_{\text{AR}} + \lambda\mathcal{L}_{\text{NCE}}$, swept over $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$
all-pinned	none (baseline)	location, scale and shape pins active; orientation held by the absence of a contrastive term
Cloc	location	distance drive with scale and shape pinned
Cscale1500	scale	scale pin retargeted to $s_0 = 1500$
Crank15	shape	shape pin retargeted to $r_0 = 15$
Corient	orientation	contrastive drive with the other three pins active
Clocorient	location and orientation	both drives with scale and shape pinned
loc-drive λ=0.1	location	location dose ladder at $\lambda_d = 0.1$: distance drive with scale held
full+orient-pinned λ=0.9	orientation	orientation dose ladder at $\lambda_o = 0.9$: contrastive drive with location and scale held

Table 4.5: The training conditions. The first block separates the contribution of each training stage, the second uses a contrastive objective, the third is single-axis factorial, where one axis is changed and the other three are kept fixed and measured, and the fourth gives the two dose-ladder rungs that appear alongside the factorial conditions in Table 4.7.

4.4 Main result

Equation (3.9)'s method is tested against the same architecture trained with the same budget, where all four axes are set to their values after Stage 1, under the all-pinned condition. This is the correct reference for comparison because the method differs only in the location drive; every other axis, and the entire training recipe, is identical. Table 4.6 represents the comparison.

	brief captioning, n=5000					dd256, n=1300				
	CLIP↑	SPICE↑	METEOR↑	BLEU-4↑	CIDEr↑	CLIP↑	CH_s↓	CH_i↓	rec.↑	obj/cap
all-pinned	0.5799	0.1003	0.1525	0.0340	$< 10^{-4}$	0.5703	0.7885	0.4396	0.4529	8.69
Cloc	0.6325	0.1212	0.1714	**0.0445**	**0.0002**	0.6187	**0.7369**	0.3698	0.5116	8.98
Clocorient	**0.6421**	**0.1279**	**0.1733**	0.0445	$< 10^{-4}$	**0.6312**	0.7638	**0.3656**	**0.5336**	8.94

Table 4.6: The method (Cloc) against the geometrically pinned baseline, with the combined variant (Clocorient) for reference. CLIP is CLIPScore, CH is CHAIR, rec. is object recall, obj/cap is objects mentioned per caption; an up arrow marks a column where higher is better, and the best value per column is in bold. The method gains +0.0526 CLIPScore on the brief set, worth 22.9 standard errors, and all three of its CHAIR differences have 95% bootstrap intervals excluding the baseline. The two n-gram metrics sit near their floor on captioners trained this briefly, so they separate the conditions by very little: the method holds the best of both, and CIDEr clears 10^{-4} for no other condition. Caption length is in Table 4.16.

The technique benefits each metric employed. Agreement between images and text without a reference increases by 0.0526 for the short dataset (22.9 standard errors) and 0.0484 for the long dataset. Object hallucination decreases in both cases: $CHAIR_s$ from 0.7885 to 0.7369, and $CHAIR_i$ from 0.4396 to 0.3698; object recall increases from 0.4529 to 0.5116, meaning that the model refers to more present objects and makes fewer mistakes with them. Objects mentioned per caption rise from 8.69 to 8.98 and captions lengthen by 5.2 words, both with intervals excluding zero, so the fall in $CHAIR_i$ is not the model saying less. The two reference-based metrics, calculated using human captions as a reference instead of learned ones, behave similarly: SPICE increased by +0.0209, and METEOR increased by +0.0189.

With an additional contrastive orientation drive, condition Clocorient obtains the highest scores for both the brief dataset and dd256 in the fully pinned category, 0.6421 and 0.6312, respectively, with the lowest $CHAIR_i$ and the highest recall. We do not use it as the algorithm of choice since it involves two drives simultaneously; the centroid distance falls from 30.97 under the location drive alone to 11.72 here, whereas subspace overlap increases from 0.0519 to 0.2993. The additional improvement of +0.0096 cannot thus be attributed to the orientation only but results from the additional constraint, and the location drive only still yields the lowest $CHAIR_s$ among all conditions. Section 4.6 comes back to this issue. The rest of this chapter establishes that each component of Equation (3.9) earns its place, and that the one axis left undriven does not repay driving it.

4.5 Ablations

The following subsections systematically deconstruct the training configuration established in Equation (3.9). Before attributing downstream performance changes to specific geometric interventions, we first verify that our diverse evaluation metrics provide a consistent, reliable ranking of the models.

4.5.1 Inter-Metric Agreement in Condition Ranking

Both evaluation sets in Table 4.7 include all eight conditions. They differ in generation budget, with sixty-four tokens and two hundred and fifty-six tokens respectively. This means they are testing different abilities: the model can produce fifty-word captions on average in one case and ninety-five-word captions in the other. There are four ranking measures drawn from two sources of ground truth and three ways of comparing against them, and they rank conditions identically. CLIPScore evaluates the caption using the image and a contrastive model and does not require any reference text; SPICE compares scene graphs extracted from candidate and reference texts; finally, METEOR compares against the reference texts through stem- and synonym-aware unigram alignment. All four produce the same ranking, which is the row order of Table 4.7: no condition changes place whether it is judged by the image, by scene graphs, or by lexical alignment. That reference-free and reference-based metrics agree with each other is useful: any ordering change that flattered CLIPScore without improving the caption would have no effect on SPICE and METEOR.

The measures that depart from it are the hallucination rates and object recall, and every departure is small in value even where it is not adjacent in rank. On dd256 CHAIR_s prefers Cloc to Clocorient by 0.0269, moving it past loc-drive, and full+orient-pinned to Corient by 0.0030; CHAIR_i places all-pinned above Corient by 0.0085, and full+orient-pinned above both; recall places Crank15 above all-pinned by 0.0066. Cscale1500 holds last place under every measure in the table, and Clocorient, loc-drive and Cloc hold the top three under every measure, exchanging order among themselves according to whether a measure rewards restraint or coverage. The families agree wherever there is a resolved effect to agree about and differ only among conditions the intervals cannot separate in the first place.

Two other aspects of this table that come into play here are worth noting. One is the spread of the data, which is quite wide: from Cscale1500 to Clocorient, CLIPScore shifts by 0.0955 for the short dataset and 0.1057 for dd256, based on baselines of 0.5799 and 0.5703 respectively, meaning that the geometric conditions shift the model's predictions by between a sixth and a fifth of the baseline score. The other is the stability of the CLIPScore ranking: no condition switches place when the model is asked for a longer caption rather than a shorter one, though CHAIR_i does reorder the middle of the table, moving Corient from the best of the three unpinned-orientation conditions to the worst.

	brief captioning, n=5000						dd256, n=1300			
Condition	CLIP↑	SPICE↑	METEOR↑	BLEU-4↑	CIDEr↑	CH_i↓	CLIP↑	CH_s↓	CH_i↓	rec.↑
Clocorient	**0.6421**	**0.1279**	**0.1733**	0.0445	$< 10^{-4}$	**0.3338**	**0.6312**	0.7638	**0.3656**	**0.5336**
loc-drive λ=0.1	0.6337	0.1217	0.1729	**0.0463**	**0.0004**	0.3587	0.6207	0.7477	0.3777	0.5216
Cloc	0.6325	0.1212	0.1714	0.0445	0.0002	0.3490	0.6187	**0.7369**	0.3698	0.5116
Corient	0.5950	0.1085	0.1573	0.0369	$< 10^{-4}$	0.4133	0.5787	0.7838	0.4481	0.4524
full+orient-pinned λ=0.9	0.5886	0.1053	0.1563	0.0358	$< 10^{-4}$	0.4188	0.5786	0.7808	0.4356	0.4680
all-pinned	0.5799	0.1003	0.1525	0.0340	$< 10^{-4}$	0.4323	0.5703	0.7885	0.4396	0.4529
Crank15	0.5769	0.0980	0.1513	0.0337	$< 10^{-4}$	0.4477	0.5673	0.7923	0.4540	0.4595
Cscale1500	0.5466	0.0898	0.1458	0.0313	$< 10^{-4}$	0.4683	0.5255	0.8900	0.5941	0.3558

Table 4.7: All eight conditions on both evaluation sets, ordered by brief-set CLIPScore, which is also the order every one of the four ranking measures produces. CLIP is CLIPScore, CH is CHAIR, rec. is object recall; an up arrow marks a column where higher is better, and the best value per column is in bold. The two n-gram columns are the exception to the common order: both are maximised by the dose-ladder rung rather than by Clocorient, they separate the eight conditions over a range of 0.0150 in BLEU-4, and CIDEr clears 10^{-4} for only the two strongest location drives. The conditions are defined in Table 4.5.

4.5.2 Structural Origins of the Modality Gap

These four baseline conditions constitute a 2×2 factorial with respect to the presence of each stage, with stage2-only and stage1-only acting as parallel stages rather than sequential ones. The text representations remain the same for all four conditions with a total variance of 0.0519, as they are derived from the frozen embedding table; any variation shown in Table 4.8 is an image cloud variation.

Condition	G_μ	$\widehat{G}_\mu$	$\operatorname{tr}\Sigma_X$	r	α_X	O_{16}
random	40.98	2.707	229.1	63.01	1.776	0.0042
stage2-only	93.82	5.902	252.7	36.90	1.790	0.0041
stage1-only	177.67	2.595	4686.6	39.19	2.336	0.0510
full	245.46	3.626	4581.6	38.61	2.341	0.0492

Table 4.8: Gap geometry by training stage, MSCOCO `val2017`, n=5000. Chance-level overlap at q=16 is $q/d \approx 0.0039$, from Equation (3.4).

Each stage owns a distinct part of the geometry. Stage 1 creates the orientation: without it the two principal subspaces overlap only at chance, 0.0042 for random and 0.0041 for stage2-only against 0.0039, and with it overlap reaches 0.0510, about 13.1 times chance. Autoregressive training alone produces no angular alignment. Stage 1 also sets scale and anisotropy, raising total variance from 252.7 to 4581.6, a factor of 18, and the spectral exponent from 1.78 to 2.34. Stage 2 creates the location gap, bringing the centroid distance from 177.67 to 245.46, an increase of 38%: the instruction objective has no alignment component and drives the representation farther away from the text manifold. Consequently, this stage creates much of the gap that the later interventions close, which is why the reference value is 177.67 instead of the drifted 245.46. Moreover, the location gap at this stage becomes a

cleaner rigid translation. The residual ratio

$$\rho_r := \frac{\operatorname{tr}\Sigma_r}{G_\mu^2 + \operatorname{tr}\Sigma_r}, \qquad \Sigma_r = \operatorname{Cov}\big[(x_i - \bar{x}) - (y_i - \bar{y})\big], \tag{4.6}$$

falls from 0.1292 to 0.0706, so the share of the gap's second moment explained by the centroid offset rises from 87.1% to 92.9%. The clouds are not tightening: $\operatorname{tr}\Sigma_r$ holds at 4684.0 against 4579.1 while G_μ^2 nearly doubles.

Condition	CLIPScore↑	SPICE↑	METEOR↑	BLEU-4↑	CIDEr↑
random	n/a	n/a	0.0188	$< 10^{-4}$	0.0002
stage2-only	n/a	n/a	0.1005	0.0132	$< 10^{-4}$
stage1-only	n/a	n/a	0.0004	$< 10^{-4}$	$< 10^{-4}$
full	0.5433	0.0908	0.1595	0.0455	0.0097

Table 4.9: Captioning by training stage, n=5000. The three degenerate conditions were scored with n-gram metrics only, so their reference-free and scene-graph entries are unavailable.

Table 4.9 also evaluates the same four conditions for captioning. Both stages are required. The stage1-only condition is aligned but was never trained to generate text from an image, and it generates virtually nothing (METEOR = 0.0004); the stage2-only condition can generate text but was never aligned, achieving a score of 0.1005. Together, they achieve 0.1595, with CLIPScore = 0.5433 and SPICE = 0.0908. Contrastive pre-training is insufficient alone, but it adds substantially on top of autoregressive training.

4.5.3 Performance Saturation Under Contrastive Scaling

Table 4.10 sweeps the contrastive weight λ. Geometry follows the dose monotonically: overlap grows by a factor of 2.8, from 12.6× to 35.0× chance, the normalised location gap falls from 3.626 to 2.420, and variance drops 20%. Caption quality does not. CLIPScore gains 0.0557 between λ=0 and λ=0.1, worth 20.7 standard errors, then flattens: across $[0.1, 0.9]$ the spread is 0.0051, or 1.9 standard errors, while orientation more than doubles. SPICE and METEOR, despite relying on a different source of truth, exhibit the same profile: both react to the initial dose and subsequently remain within the intervals of $[0.1091, 0.1158]$ and $[0.1702, 0.1785]$. CIDEr stands out as a complete outlier here and offers little practical value. It swings by a full order of magnitude while the other scores barely move; consequently, we only include it for the sake of completeness rather than relying on it as a primary metric.

The table also shows why this design cannot say *which* change is responsible. Raising λ lifts orientation, closes location, compresses scale and raises rank at once, and although the objective is nominally angular it still moves the centroid by a third of its initial value. With four candidate causes moving together, no single-axis correlation is interpretable.

Condition	λ	O_{16}	O_{64}	$\widehat{G}_\mu$	$\operatorname{tr}\Sigma_X$	CLIP↑	SPICE↑	METEOR↑	BLEU-4↑	CIDEr↑
full	0.0	0.0492	0.0848	3.626	4581.6	0.5433	0.0908	0.1595	0.0455	0.0097
full+orient	0.1	0.0623	0.0931	3.454	4486.6	0.5990	0.1091	0.1749	0.0541	0.0374
full+orient	0.3	0.0817	0.1089	3.279	4308.8	0.6008	0.1129	0.1747	0.0507	0.0037
full+orient	0.5	0.0971	0.1257	3.116	4046.0	0.6036	0.1137	**0.1785**	0.0562	0.0258
full+orient	0.7	0.1110	0.1422	2.841	3779.5	0.6033	0.1122	0.1780	**0.0584**	**0.0579**
full+orient	0.9	0.1369	0.1729	2.420	3687.5	**0.6041**	**0.1158**	0.1702	0.0453	0.0010

Table 4.10: full+orient: contrastive dose sweep, n=5000. CLIP is CLIPScore. Every geometric quantity moves monotonically with λ; the quality metrics do not. BLEU-4 makes the point sharply: it peaks at $\lambda = 0.7$ and then falls to 0.0453 at $\lambda = 0.9$, below the 0.0455 of the run with no contrastive term at all, so the strongest dose of the objective that moves orientation furthest is the one that buys nothing. Effective rank rises from 38.61 to 42.56 across the sweep and is omitted for width.

4.5.4 Objective Targeting: `[CLS]` versus Pooled Representations

Section 3.3 evaluates each axis of the pooled vector. This step is necessary because the decoder processes and the metrics ultimately measure the mean of the 257 projected image tokens. An objective function, on the other hand, can target a completely different vector. The projected `[CLS]` token serves as the most natural candidate here. It provides a single vector rather than an averaged representation, making it exactly the kind of input a contrastive objective typically expects. Table 4.11 illustrates this by applying the identical distance objective to both approaches.

The difference is not one of degree. At the same weight $\lambda = 0.1$, acting on `[CLS]` moves the centroid distance from 245.46 to 210.64, a reduction of 35; acting on the pooled vector moves it to 24.03, a reduction of 221. The objective is identical and the optimisation budget is identical. Only the vector it is written against differs, and the axis it is meant to control moves six times as far in one case as in the other.

Raising the dose does not repair it. At $\lambda = 0.5$ on `[CLS]`, with the scale pin active, the centroid reaches 192.75 and CLIPScore *falls*, from 0.5847 to 0.5780. More of a control that acts on the wrong object is worse than less of it, because the additional weight is taken from the language objective, which the model does need, and spent on a quantity the decoder never reads.

The pooled rows also show why the scale pin belongs to the method rather than decorating it. Driving the distance on the pooled vector without the pin gives the highest CLIPScore in the table, 0.6481, but it does so while total variance collapses from 4581.6 to 1723.0, a loss of 62%, and the participation ratio rises to 45.98. The distance term shrinks the cloud while it moves it, because contracting toward the target reduces the centroid offset and the spread at the same time. Three axes move together, so that row cannot carry an attribution however well it scores. With the pin, the trace is held at 4791.9 and CLIPScore is 0.6337: the pin costs 0.0144 of CLIPScore and buys back the scale axis. Shape is not yet severed: the participation ratio still falls to 21.30, which is what the rank pin is for.

Cond.	Acts on	λ	G_μ	$\operatorname{tr}\Sigma_X$	r	CLIP↑	SPICE↑	METEOR↑
full	none	n/a	245.46	4581.6	38.61	0.5433	0.0908	0.1595
loc-drive	`[CLS]`	0.1	210.64	4087.0	39.46	0.5847	0.1032	0.1669
loc-drive	`[CLS]`, pin	0.5	192.75	4563.5	38.00	0.5780	0.0978	0.1632
loc-drive	pooled	0.1	24.03	1723.0	45.98	**0.6481**	**0.1276**	**0.1779**
loc-drive	pooled, pin	0.1	29.61	4791.9	21.30	0.6337	0.1217	0.1729

Table 4.11: The same objective applied to two different vectors, n=5000. "pin" is the scale pin; CLIP is CLIPScore.

4.6 Factorial Analysis of Geometric Interventions

Table 4.12 verifies that each intervention moved its own axis and left the other three where the baseline put them.

Condition	Axis released	G_μ	$\operatorname{tr}\Sigma_X$	r	O_{16}
all-pinned	none (all pinned)	177.86	4252.7	36.99	0.0515
Cloc	location	**30.97**	4789.6	37.32	0.0519
Cscale1500	scale	177.74	**1487.6**	38.67	0.0435
Crank15	shape	177.81	4311.6	**20.21**	0.0517
Corient	orientation	177.61	4578.8	35.95	**0.1114**
Clocorient	location and orientation	**11.72**	4804.1	36.32	**0.2993**

Table 4.12: Single-axis isolation, n=5000. Bold entries are the targeted axis; the other columns move by at most 16% against on-axis moves of 45% to 116%, so the isolation is verified from measurements rather than assumed.

Still, the isolation property is satisfied because the multiplication of the displacement factor by the constant 5.7 is done by Cloc, while rank and overlap hold to within 1% and variance rises 12.6%, all small against the 83% move in the centroid. The variance is reduced to 35% with the help of Cscale1500, and the centroid position is maintained up to 0.1%. The rank is reduced from 37.0 to 20.2 using Crank15. The value of overlap is doubled using Corient, with the centroid unchanged and variance and rank within 8%. Figure 4.1 shows the same six conditions geometrically.

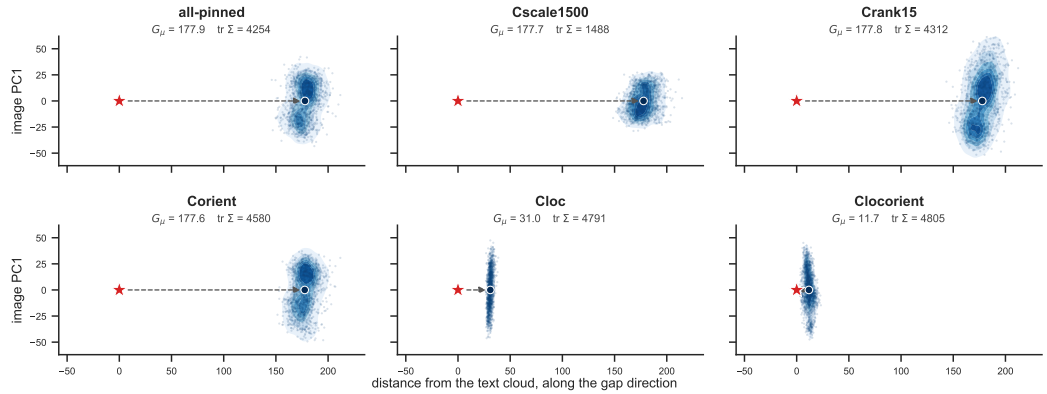


Figure 4.1: Where each intervention leaves the image cloud. Each panel holds the pooled image embeddings of one condition, n=1300, projected onto that condition's own gap direction (horizontal) and the leading direction of its variance orthogonal to it (vertical). The red star is the text cloud, which is identical in every panel because the text tower is frozen, and the dashed arrow is the centroid distance G_μ. Scale, shape and orientation leave the cloud where the baseline put it, at 177.6 to 177.9; only Cloc and Clocorient move it, to 30.97 and 11.72. Cscale1500's cloud is visibly smaller, which is its $\operatorname{tr}\Sigma_X$ of 1487.6 against roughly 4300 elsewhere. Shading is a kernel density estimate over the same points.

	brief captioning, n=5000						dd256, n=1300			
Cond.	CLIP ↑	z ↑	SPICE ↑	METEOR ↑	BLEU-4 ↑	CIDEr ↑	CLIP ↑	CH_s ↓	CH_i ↓	rec. ↑
all-pinned	0.5799	n/a	0.1003	0.1525	0.0340	$< 10^{-4}$	0.5703	0.7885	0.4396	0.4529
Cloc	0.6325	+22.9	0.1212	0.1714	**0.0445**	**0.0002**	0.6187	**0.7369***	0.3698*	0.5116*
Cscale1500	0.5466	−12.8	0.0898	0.1458	0.0313	$< 10^{-4}$	0.5255	0.8900*	0.5941*	0.3558*
Crank15	0.5769	−1.3	0.0980	0.1513	0.0337	$< 10^{-4}$	0.5673	0.7923	0.4540*	0.4595
Corient	0.5950	+6.3	0.1085	0.1573	0.0369	$< 10^{-4}$	0.5787	0.7838	0.4481	0.4524
Clocorient	**0.6421**	**+27.4**	**0.1279**	**0.1733**	0.0445	$< 10^{-4}$	**0.6312**	0.7638	**0.3656***	**0.5336***

Table 4.13: Effect of each intervention on both evaluation sets. CLIP is CLIPScore, CH is CHAIR, rec. is object recall. z is against all-pinned on the brief-set CLIPScore; an asterisk marks a CHAIR value whose 95% bootstrap interval excludes the baseline. Lower CHAIR is better, higher everything else. The two n-gram columns agree with the rest on which conditions help and which hurt, but over a far narrower range: BLEU-4 spans 0.0313 to 0.0445 across all six, and CIDEr stays below 10^{-4} everywhere except Cloc.

Table 4.13 reports the downstream effect of each intervention. The four axes behave differently, and the metric families agree on the pattern.

Location is a lever (Cloc). Closing the centroid raises CLIPScore by 0.0526 ($z = 22.9$) on the brief set and 0.0484 on dd256, SPICE by 0.0209, METEOR by 0.0189, and improves both CHAIR measures and object recall at once, the last by 0.059. It does not do this by saying less. On brief captions the closed condition names *fewer* objects, 4.60 against 4.88, in shorter text, yet recalls more of those present (0.4528 against 0.3875) and hallucinates a smaller *fraction* of what it names (0.3490 against 0.4323). On detailed descriptions the count reverses, 8.98 against

8.69, while recall and hallucination move the same way. It buys precision under a short budget and coverage under a long one; reticence explains neither.

Scale is a mirror (Cscale1500). Variance compression reverses everything: CLIPScore goes down by 0.0333 ($z = -12.8$) for the short set and 0.0448 for dd256, SPICE to 0.0898 and METEOR to 0.1458, both CHAIR scores deteriorate, recall decreases by 0.097, and captions get shorter, going down from 94.9 to 76.7 words. The process can be seen in the texts themselves. For the detailed set, 92.3% of captions generated under this condition start with "nobody is" compared to 0% from the baseline; thus, not only does variance compression lower the quality of the descriptions, but it forces the decoder to choose a degenerate starting phrase. Part of the CHAIR penalty follows from that opener directly, since the object named inside a negation still counts as a mention.

Shape is a null (Crank15). Effective rank halving makes caption quality change by nothing significant, $z = -1.3$. SPICE and METEOR differ from the baseline by only 0.0023 and 0.0011. Only one effect is resolved and negative: CHAIR_i increases by 0.0144, with an interval of $[0.0024, 0.0269]$ that does not include the baseline, along with an increase of 0.48 objects per caption and 3.1 words. The increased concentration of the variance causes the model to name more objects and recover no more, which is a failure of moderation rather than quality; it is not the way a useful axis would behave.

Orientation is weakly effective (Corient). Doubling overlap increases CLIPScore by 0.0151 ($z = 6.3$), which is about one-third the amount of the location effect and does not take any of the metrics past their intervals; SPICE and METEOR improve by 0.0082 and 0.0048, respectively, compared to 0.0209 and 0.0189 for location. Orientation is simple to *manipulate* but weak in *effect*, a distinction that the sweep in Section 4.5.3 could not have drawn. The same caution as for Cscale1500 applies in milder form: 36.3% of this condition's detail captions open with a degenerate fragment, against none under location closure, so its weak gain is measured on a decoder that is partly unstable.

The families agree on the ordering, and the exceptions are informative. Four measures rank the six conditions identically, CLIPScore on both sets together with SPICE and METEOR:

$$\text{Clocorient} > \text{Cloc} > \text{Corient} > \text{all-pinned} > \text{Crank15} > \text{Cscale1500}.$$

The others switch adjacent pairs. BLEU-4 and CHAIR_s rate Cloc over Clocorient, CHAIR_i ranks the baseline over Corient, and recall rates Crank15 over the baseline. All switches are within the noise: in each case the CHAIR interval for the switching effect contains the baseline, while Cloc and Clocorient only differ in BLEU-4 by 5×10^{-6}. Cloc, Clocorient, and Cscale1500, the three conditions with the resolved

CHAIR effects, rank equally high under all metrics. The metric families concur on those cases.

Mirror and Null give it its nature as an attribution. As an independent finding, the location result faces the challenge of being open to criticism that any disruption is bound to interfere with the decoder, and the benefit comes as a by-product. The fact that Cscale1500 causes similar disruption but makes everything much worse, while Crank15 disrupts substantially and leaves caption quality unchanged, indicates that the critical issue is the motion of the axis.

When the Clocorient combines the two drives, the location and orientation drives, the highest values of the fully pinned family appear, that is, 0.6421 ($z = 27.4$), with the minimum of CHAIR_i and the maximum recall. However, it does not indicate which orientation contributes, since it does not control for location: the distance to the centroid decreases from 30.97 when the location drive works alone to 11.72 in this case, and the overlap increases from 0.0519 to 0.2993. Both axes move at the same time, and therefore the gain of +0.0096 above the location drive alone cannot be attributed to the orientation, but to the additional closure that goes along with it. The location drive alone also has the best CHAIR_s of any condition, at 0.7369 compared to 0.7638. That is why the method from Equation (3.9) moves only one axis.

4.6.1 Training-time verification

Table 4.12 verifies the isolation once, on the evaluation set. Figures 4.2 to 4.5 verify it at every step, on the batches the objective actually saw, one figure per axis with all six conditions on each. Figure 4.6 then checks that none of it was bought by damaging the decoder. Every condition ran 450 steps at an effective batch of 32, and the geometry leg sees 64 images per step.

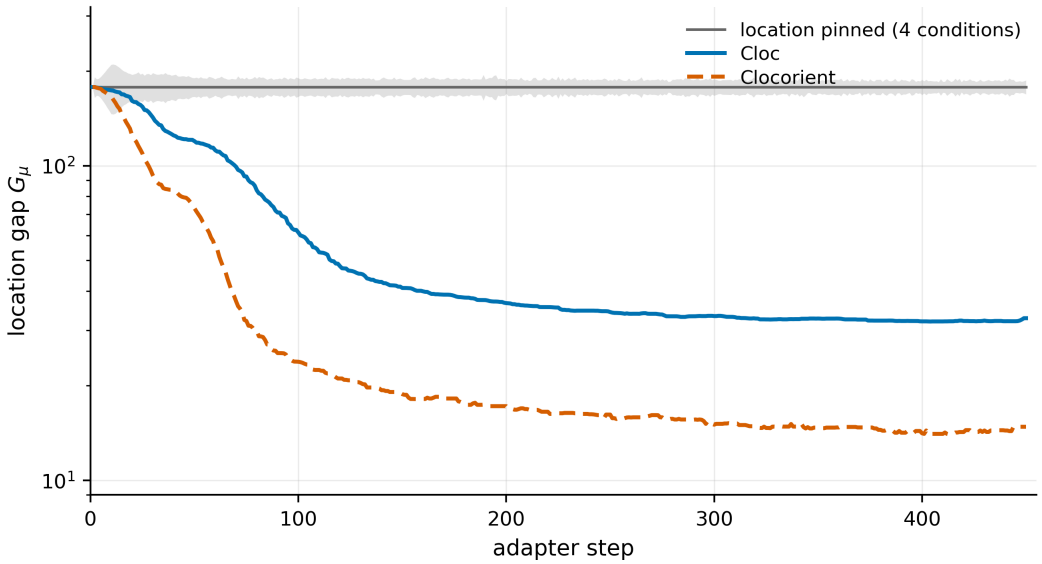


Figure 4.2: Location. Cloc and Clocorient log the gap to the text centroid directly and drive it from 177.9 to 32.8 and 14.8. The four conditions that pin location log only displacement from their own frozen centroid, so their gap is not measured; the band is the interval the triangle inequality allows, 177.86 plus or minus that displacement, and it never approaches the two driven conditions.

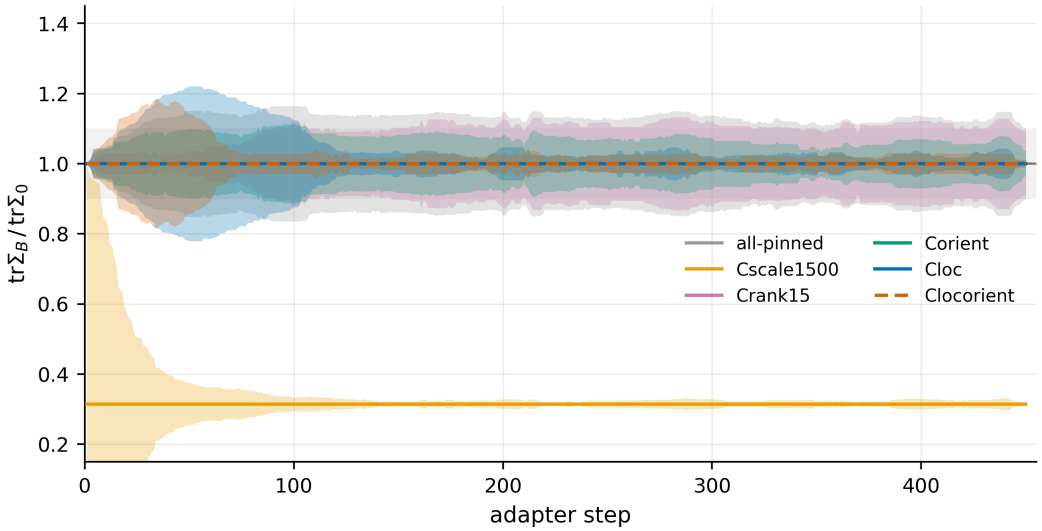


Figure 4.3: Scale, as total variance divided by the common baseline $\mathrm{tr}\,\Sigma_0 = 4767.3$, auto-measured at the first step. A pinned axis sits on 1 and the shaded band is $\pm 10\%$. Five conditions hold there; only Cscale1500 leaves, settling at 0.32, which is its target of 1500 against that baseline. The quantity is reconstructed from the squared pin term, which loses the sign of the deviation, so it is drawn as a band rather than a line.

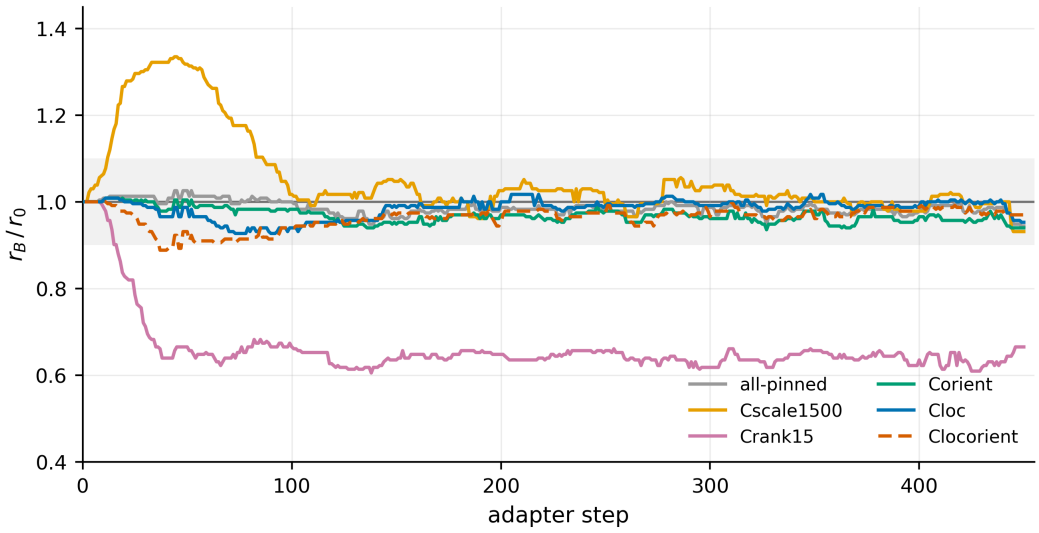


Figure 4.4: Shape, as the batch participation ratio divided by the common baseline $r_0 = 23.30$. Five conditions hold at 1; only Crank15 leaves, settling at 0.64, which is its target of 15 against that baseline. Cscale1500's excursion over the first hundred steps returns to the band. This ratio is logged directly, so no reconstruction is involved.

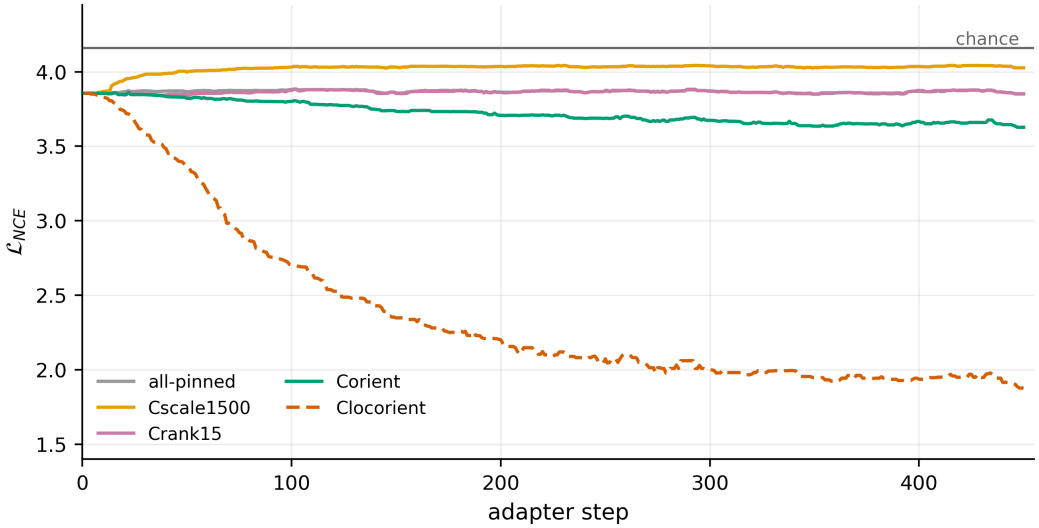


Figure 4.5: Orientation, as the contrastive term the orientation drive optimises, with chance for a 64-image batch marked at $\ln 64$. The conditions that pin orientation sit just under chance and stay there; Corient moves it to 3.63 and Clocorient, which carries the same dosage alongside the location drive, to 1.88. Cloc is absent because the distance trainer has no contrastive leg and never evaluates this term; its evaluated overlap is at baseline, 0.0519 against 0.0515, so orientation is untouched there.

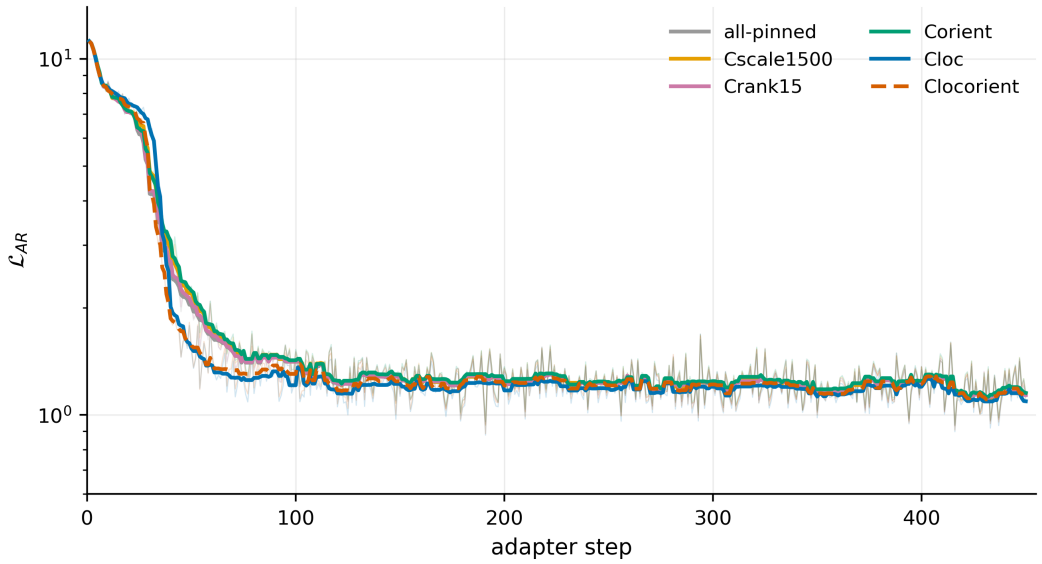
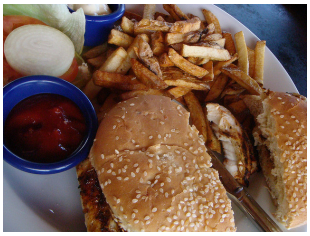


Figure 4.6: The autoregressive loss over the same runs, on a logarithmic axis, per step in light and as a 15-step rolling median in bold. Every condition falls from its initial value and settles together, so none of the geometric terms is bought by degrading the language objective.

The evaluated G_μ, total variance, participation ratio and subspace overlap for all six conditions, measured on the full set rather than per batch, are in Table 4.12.

4.6.2 Qualitative examples

The overall statistics represent averages for the entire dataset. Here are five examples that demonstrate how the axes affect individual captions. These five captions come from the detailed dataset, where the generation budget is sufficiently large so that each caption ends individually.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a **man** standing in a kitchen, holding a large, brown, and furry **dog**. The **dog** is sitting on a countertop, and the **man** is holding it in his arms. The countertop is covered with various food items, including a bowl of cereal, a plate of eggs, and a cup of coffee. There are also several other food items scattered around the kitchen, such as a **bottle** of ketchup, a **bottle** of mustard, and a **bottle** of hot sauce. The **man** and the **dog** appear to be enjoying their meal together in the kitchen.	0.471	9	2
Cscale1500	nobody is sitting at a table, but there is a **chair** and a plate on the floor. The **chair** is positioned near a wall, and the plate is on the floor next to the **chair**. There are also two cups on the floor, one on the left side and the other on the right side. The scene appears to be a messy room with a **chair** and a plate on the floor, and two cups nearby.	0.654	4	2
Crank15	The image features a **man** standing in a kitchen, holding a large, brown, and furry **dog**. The **dog** is sitting on a countertop, and the **man** is petting it. There are several other objects in the kitchen, including a bowl, a **spoon**, and a **bottle**. The scene is set in a well-lit room, with a window visible in the background. The **man** and the **dog** appear to be enjoying their time together in the kitchen, creating a cozy and relaxed atmosphere.	0.431	8	1
Corient	nobody is eating a sandwich in the image, but there are two sandwiches on the table. The sandwiches are placed on a plate, and there are two cups of soda on the table as well. The table is located in a kitchen, and there are several other items on the table, such as a **bottle**, a glass, and a **spoon**. The scene appears to be a casual dining setting, where **people** are enjoying a meal together.	0.711	3	3
Cloc ⋆	The image features a large, colorful, and delicious-looking sandwich, which is placed on a plate. The sandwich is made up of a hamburger patty, cheese, and a slice of tomato, all of which are placed on a bun. The sandwich is served with a side of fries, which are placed on a separate plate. The sandwich is surrounded by various condiments, including ketchup, mustard, and pickles, which are placed on the plate. There are also two slices of tomato on the plate, one of which is placed on top of the sandwich and the other on the side. The sandwich is served with a drink, likely a soda, which is placed on the table.	0.859	0	2
Clocorient ⋆	The image features a large, colorful, and delicious-looking sandwich, placed on a plate. The sandwich is made up of a hamburger patty, which is placed on top of a bun, and is surrounded by various toppings. The sandwich is served on a plate, and there are two slices of tomato on the plate, one on top of the sandwich and the other on the side. The sandwich is accompanied by a bowl of fries, which are placed on the plate next to the sandwich. There are also two cups of soda, one on the left side of the plate and the other on the right side. The entire meal is presented in a visually appealing manner, with the sandwich and fries standing out against the white plate.	0.770	0	3

Figure 4.7: A sandwich and fries. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The baseline and the shape drive both describe a man holding a furry dog in a kitchen, spending nine and eight words on objects that are not there, and the scale and orientation drives open on a fixed fragment. Both driven conditions describe the plate itself, a sandwich on a bun beside a portion of fries, and invent nothing. Cloc reaches 0.859 against the baseline's 0.471, the largest gain in these examples.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a group of horses standing in a grassy field. There are two horses in the front, and two more horses in the back. The horses are grazing and appear to be enjoying the grass. The scene is set against a blue sky, and the horses are surrounded by greenery. There are no other animals or **people** in the scene.	0.563	1	1
Cscale1500	nobody is sitting on the **couch**, but there are two **people** standing on it. One **person** is standing in the middle of the **couch**, while the other **person** is standing near the edge. Both **people** are facing the camera. The **couch** is located in a living room, and there are several other items present in the room, such as a **television**, a **chair**, and a **table**. The room appears to be well-furnished and comfortable. There are several **books** scattered around the room, some of which are visible on the **couch** and on the floor. The **books** appear to be a mix of fiction and non-fiction titles.	0.347	6	0
Crank15	The image features a group of horses standing in a grassy field, with one horse standing out from the group. The horses are grazing and looking in different directions, creating a natural and relaxed atmosphere. The scene is set against a blue sky, with a few clouds visible in the background. The horses are located in a field with a fence in the distance, providing a sense of openness and freedom. There are several other horses in the field, some of which are further away from the camera. The horses appear to be healthy and well-cared for, as they are grazing peacefully in the field.	0.563	0	1
Corient	The image features a group of horses grazing on a grassy field. There are several horses in the scene, some of which are standing and others are lying down. The horses are scattered throughout the field, with some closer to the center and others towards the edges. The scene is surrounded by a fence, which can be seen in the background. The fence is made of wooden planks, and it appears to be a natural barrier for the horses to graze and roam freely. The fence also provides a sense of security for the horses, as it helps to keep them contained within the field.	0.587	0	1
Cloc ⋆	The image features a group of zebras standing in a grassy field, with a few of them facing the camera. There are a total of 12 zebras in the scene, with some of them standing closer to the camera and others further away. The zebras are arranged in a semi-circle formation, with some of them facing each other. The zebras are grazing on the grass, and there are several other animals in the background, such as a horse and a donkey. The scene is set in a natural environment, with the zebras and other animals roaming freely in the field.	0.830	0	2
Clocorient ⋆	The image features a large group of zebras grazing on a grassy field. There are a total of 12 zebras in the scene, with some of them standing close together and others scattered throughout the field. The zebras are of various colors, including black, white, and brown. The zebras are surrounded by a fence, which is visible in the background. There are also some trees in the background, adding to the natural setting. The grassy field is well-maintained, with the zebras grazing and moving around freely.	0.820	0	1

Figure 4.8: Zebras grazing. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. Every condition that leaves the gap open calls the animals horses, and compressing the scale abandons the scene for a living room. Both driven conditions name the zebras and score above 0.82, against 0.56 for the baseline, the largest margin in these examples.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a large, well-lit room with a **desk** and a **chair** in the center. There are several **books** on the **desk**, and a **computer** is placed on the left side of the **desk**. The room is spacious and has a large window, allowing natural light to enter. In the room, there are two people, a man and a woman, sitting at the **desk**. The man is wearing a suit and **tie**, while the woman is wearing a dress. They are both working on their **computers**, with the man using a **laptop** and the woman using a desktop **computer**. There are several other people in the room, some of whom are standing and others sitting. Some of them are engaged in conversations, while others are looking at the **computer** screens or reading **books**. The room appears to be a busy and productive workspace.	0.388	13	1
Cscale1500	nobody is sitting at the **table**, but there are two **chairs** placed on either side of the **table**. The **table** is set with a white tablecloth, and there are two **cups** on the **table**. One **cup** is placed on the left side of the **table**, while the other **cup** is placed on the right side. Additionally, there is a glass of water on the **table**, which is likely for the **cups** to be filled with. There are two **books** on the **table**, one placed on the left side and the other on the right side. The **books** appear to be placed there for the person to read or reference them during the meal.	0.447	14	1
Crank15	The image features a large, well-maintained, and beautifully decorated house with a large front door. The house is located in a residential neighborhood, surrounded by trees and other houses. The front door is adorned with a large, colorful wreath, adding a festive touch to the scene. The house is also decorated with a Christmas tree, which is visible inside the front door. The house appears to be well-lit, with lights illuminating the front yard and the surrounding trees. There are several people in the image, some of whom are standing near the front door, while others are further away. The people appear to be engaged in conversation, possibly discussing the festive decorations or the holiday season.	0.284	0	1
Corient	nobody is visible in the image, but there are several **books** on a **table**. The **books** are arranged in a neat and organized manner, with some placed on the **table** and others stacked on top of the **table**. The **books** appear to be a mixture of fiction and non-fiction titles, with a variety of colors and sizes. There are also several **chairs** in the room, with one **chair** placed near the **table** and another **chair** further away. The room has a wooden floor, and the **chairs** are likely made of wood as well. The overall atmosphere of the room is one of organized study and reading.	0.370	11	0
Cloc $\star$	The image features a large, ornate clock tower located in a church or cathedral. The clock tower is situated on the top of a building, with a large clock face visible on the front. The clock face is adorned with intricate details, including a large hand and a smaller hand, both pointing to the time. The clock tower is surrounded by a large, ornate building, with several windows and a balcony visible on the upper levels. The building is well-maintained and appears to be in good condition. The clock tower and the building are situated in a lush, green area, with trees and other vegetation surrounding the area.	0.632	0	1
Clocorient $\star$	The image features a large clock tower, located in a city center, with a clock face visible on the top. The clock tower is surrounded by a large building, which is likely a church or a government building. The clock tower is situated in a busy city street, with people walking and standing around the area. There are several people in the scene, including a man standing in front of the clock tower, a woman standing near the clock, and a group of people walking on the sidewalk. The clock tower is a prominent landmark in the city, and it serves as a focal point for the area.	0.651	0	2

Figure 4.9: A clock tower. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked $\star$. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The captions run longer here and are set smaller so that each is shown in full. The baseline, the scale drive and the orientation drive all move the scene indoors, spending 13, 14 and 11 words on objects that are not there, and only the two driven conditions name the clock. The shape drive is the cautionary case: its caption is wrong throughout, a decorated house rather than a tower, yet it counts no hallucinated object, because house, wreath and Christmas tree are not annotated categories, and only the reference-free score registers the error.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a room with a desk and a **chair**. The desk is located in the middle of the room, and a **chair** is placed near the desk. There is a book on the desk, and a pen is placed on the **chair**. The room is well-lit, with a lamp placed on the desk. The room also has a window, which allows natural light to enter the space. The room is neat and tidy, with the desk and **chair** placed in an organized manner.	0.628	1	2
Cscale1500	nobody is sitting at the table, but there is a **chair** and a plate with a slice of **pizza** on it. The **chair** is placed in the middle of the room, and the **pizza** is on the table. The room is dimly lit, and there is a lamp on the table. The room also has a window, which is partially visible in the image.	0.591	2	1
Crank15	The image features a man sitting at a desk, reading a book. He is holding a pen in his right hand and is wearing a black shirt. The desk is located in a room with a window, and there is a **clock** on the wall. The room is well-lit, with a lamp on the desk and a light fixture above the window. The man is sitting in a **chair**, which is positioned near the desk. There is a **chair** on the left side of the room, and a **chair** on the right side of the room. The room also has a table, which is located near the desk.	0.749	2	3
Corient	The image features a room with a large, comfortable bed, a desk, and a **chair**. The bed is situated in the center of the room, and it is surrounded by a desk and a **chair**. The desk is placed near the bed, and it is equipped with a **computer monitor**, a **keyboard**, and a **mouse**. The **chair** is positioned near the desk, providing a comfortable seating option for the person using the desk. The room is well-lit, with a lamp placed on the desk and a window in the background. The room also has a bookshelf, which is filled with books, adding to the cozy and functional atmosphere.	0.578	5	4
Cloc $\star$	The image shows a man sitting at a desk, reading a book. He is wearing a black shirt and is holding a book in his hands. The book appears to be a hardcover book with a red cover, and the man is reading it with interest. The desk is located in a room with a window, and there is a lamp on the desk, providing light for the man to read. The room is spacious and well-lit, creating a comfortable environment for the man to focus on his reading.	0.781	0	3
Clocorient $\star$	The image shows a man sitting at a desk, reading a book. He is holding a book in his hands, and the book is open on the desk. The man is wearing a black shirt and is seated in front of a window, which is partially visible in the background. There are several books scattered around the desk, with some of them placed on the desk and others on the floor. The room is well-lit, with a lamp on the desk and a window in the background.	0.783	0	3

Figure 4.10: Reading at a desk. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked $\star$. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The baseline describes an empty room and misses the person entirely. Both driven conditions produce the cleanest captions in this set, a man seated at a desk reading, with no invented object and no invented count, and they score 0.78 against the baseline's 0.63.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a white and brown **dog** sitting on a wooden dock, with its head resting on its front paws. The **dog** is facing the camera, and its body is partially submerged in the water. The dock is located on a lake or a river, with a **boat** visible in the background. The **dog** appears to be enjoying its time on the dock, possibly taking a break from swimming or playing in the water.	0.515	4	0
Cscale1500	nobody is sitting on the **couch**, but there is a **person** standing on the **couch**. The **person** is holding a white **cat**, which is sitting on the **couch**. The **cat** is facing the **person**, and the **person** is looking at the **cat**. The **couch** is located in a living room, and there is a **television** set in the room. The room is well-lit, and the **cat** is visible in the image.	0.455	13	0
Crank15	The image features a white and brown **dog** standing on a wooden dock, with its head and front legs in the water. The **dog** is facing the camera, and its body is partially submerged in the water. The **dog** appears to be enjoying its time in the water, as it is not afraid of the water and seems to be having fun. There are several **people** in the background, some of whom are standing on the dock and others are sitting on the shore. The scene is set in a wooded area, with trees and greenery visible in the background.	0.463	4	0
Corient	nobody is swimming in the water, but there are two **people** standing on a dock. One **person** is standing on the left side of the dock, while the other **person** is standing on the right side. Both individuals are facing the water, and they appear to be looking out at the water. There are several **boats** in the water, some of which are closer to the dock than others. The water is calm, and the **boats** are floating in it.	0.396	5	0
Cloc ⋆	The image features a white polar bear lying on a snowy surface, with its head resting on a snow-covered rock. The bear is positioned in a relaxed pose, with its eyes closed and its mouth open. The snow-covered rock provides a comfortable resting spot for the bear, and the surrounding snowy environment adds to the serene atmosphere. The bear's white fur contrasts with the snow, creating a visually striking composition.	0.780	0	1
Clocorient ⋆	The image features a polar bear swimming in a river, with its head and body partially submerged in the water. The bear is swimming in a river, and its head is visible above the surface. The river is flowing, and the bear is swimming in the direction of the camera. There are several **people** standing on the riverbank, observing the polar bear. Some of them are closer to the water, while others are further away. The **people** appear to be watching the bear with interest and curiosity. The scene is set in a natural environment, with trees and grass visible in the background.	0.660	2	1

Figure 4.11: A polar bear on a rock. Conditions that leave the location gap open are listed first, the two driven by this thesis last and marked ⋆. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The single annotated object is the bear, and no condition that leaves the gap open recovers it: two see a dog on a dock, one a cat on a couch indoors and one only people beside the water. Both driven conditions name the bear, although neither reads the setting correctly, one adding snow and the other a river with onlookers.

4.6.3 Location dose–response

The factorial establishes that location is the effective axis. It does not say how much closure is needed. Table 4.14 takes four points along the location ladder, from the unpinned model through the pin to two doses of the drive.

The relation holds true in both columns and is monotonic. The distance between the centroids decreases from 245.46 → 177.81 → 80.59 → 29.61, or eight-fold, while the CLIPScore goes up at each step: 0.5433 → 0.5797 → 0.6240 → 0.6337, by 14.1, 18.7, and 4.3 standard errors correspondingly. SPICE, which uses human reference captions rather than a contrastive model, follows the same progression: 0.0908 → 0.0999 → 0.1182 → 0.1217.

There are rapidly diminishing gains to be had. The first two moves gain 0.0364 and 0.0443; the third move, which reduces the centroid distance a further 2.7 times, from 80.59 to 29.61, gains only 0.0097. Most of the possible gain is realized by the point where the difference is reduced to a third of its original value, and further reduction is almost without effect. The process does not need the centroid to be driven to zero, as Section 4.5.3 showed with the same saturating shape.

Two caveats. First, the scale pin only works on the bottom three rungs and keeps total variance within 11%, that is, from 4312.3 to 4791.9, and hence, scale is not the shifting variable. Second, the participation ratio is not fixed on this particular ladder: it decreases from 38.61 → 32.32 → 27.45 → 21.30. Shape and location thus co-move on this particular ladder, and the ladder alone cannot separate the two. The factorial can, and does. Crank15 moves the participation ratio from 36.99 to 20.21, a comparable displacement, with location held at 177.81, and produces no measurable change in caption quality; the method condition Cloc, which pins shape as well as scale, holds the participation ratio at 37.32 against the baseline's 36.99 while closing the centroid, and reproduces the gain. The co-movement in this table is therefore not read as a second cause.

Condition	G_μ	$\mathrm{tr}\,\Sigma_X$	r	CLIPScore ↑	z (step) ↑	SPICE ↑
full (unpinned)	245.46	4581.6	**38.61**	0.5433	n/a	0.0908
loc-pinned	177.81	4312.3	32.32	0.5797	+14.1	0.0999
loc-drive λ=0.01	80.59	4788.0	27.45	0.6240	+**18.7**	0.1182
loc-drive λ=0.1	**29.61**	**4791.9**	21.30	**0.6337**	+4.3	**0.1217**

Table 4.14: Location dose–response, n=5000. z is against the preceding row. The scale pin is active in the lower three conditions, holding variance within 11%.

4.7 Ablating the Alignment Target: Structural vs. Semantic Anchors

The location drive moves the image center towards the average of the embedding-table lookups. The target is a feature of the input space of the decoder, not the output of some semantic encoder; it is simply where the tokens of LLaMA live, and does not

contain information regarding any specific image. Thus, it makes sense to question whether the value of the ablation in Section 4.6 lies in the geometry of the move, or in the meaning of what is being moved towards.

To distinguish the two, we swap the former for a semantically guided one, while making no other modifications. This alternative is the text tower of the same `openai/clip-vit-large-patch14` model whose vision tower provides the visual features; this model is contrastively trained to position the caption next to its own image, using a semi-orthogonal transformation W that projects from 768 to 4096 dimensions and satisfies $W^\top W = I$. Being semi-orthogonal, the lift preserves norms and inner products, so CLIP's relative geometry is transplanted intact and no trainable capacity is added to the comparison. The anchor is a loss target only and is never fed to the decoder, which continues to read LLaMA embeddings throughout. Only Cloc and Corient consume a text anchor; the remaining conditions pin image-side statistics, so for them the swap would be a no-op.

Location-wise, the swap makes no difference. The score for CLIPScore drops from 0.6187 to 0.6170 for a difference of 0.0017, where the standard error is approximately 0.003, and that of CHAIR_s increases from 0.7369 to 0.7423. Two text encodings that do not have common training data, architecture, or vocabularies move the image cloud to the same place as far as the decoder is concerned. It is only important that the cloud moves inward, not what it moves towards, and the benefit of closing the location gap is anchor-agnostic.

For orientation the swap is actively harmful, and the geometry explains why. With the embedding-table anchor, Corient lifts the top-4 overlap with the LLaMA text subspace from 0.0318 at baseline to 0.1404. With the CLIP anchor, the same objective at the same dose drives that overlap down to 0.0198, below the pinned baseline. The contrastive term still rotates the cloud, but it rotates it away from the subspace the decoder actually reads, and the captions follow: CLIPScore falls from 0.5787 to 0.5682 and CHAIR_s rises to 0.8123, worse than every condition in Table 4.13 except Cscale1500.

The orientation dimension was the only dimension along which content may have played a role, because that is the dimension over which a contrastive objective has the most direct control. Together, the two changes answer the question for both dimensions. But orientation certainly remains a valid, albeit weak, tool, as Section 4.6 showed independently; it simply shows that whatever orientation accomplishes, it does so without the help of semantics. Table 4.15 reports both swaps.

Condition	Alignment target	CLIPScore (dd256) ↑	z ↑
all-pinned	none	0.5703	n/a
Cloc	embedding-table mean	**0.6187**	n/a
Cloc_clip	frozen CLIP text tower	0.6170	**−0.4**
Corient	embedding-table mean	0.5787	n/a
Corient_clip	frozen CLIP text tower	0.5682	−2.1

Table 4.15: Alignment-anchor ablation, dd256, n=1300. z is against the corresponding embedding-table condition.

4.8 Distance to a fully-resourced reference

These varying sets of conditions do not relate to each other by more than one dimension, providing an answer to the attribution problem but nothing concerning the extent to which any of them utilize the attainable quality.

4.8.1 The reference

To quantify this in an absolute manner, we apply LLaVA-1.5-7B [9] to the same dd256 framework for inference only. It possesses similar architectural decisions: a MLP connector with two layers and a decoder from the LLaMA-2 generation, but differs from the thesis in every dimension that is advantageous: input images of 336 pixels and 576 visual tokens versus 224 and 257 for the thesis, an instruction-tuned Vicuna decoder [44] versus a base one, and training on about 1.2M samples with a fine-tuned decoder. It is a reference, not a competitor.

	steps	CLIPScore↑	CHAIR_s↓	CHAIR_i↓	recall↑	words
all-pinned (baseline)	450	0.5703	0.7885	0.4396	0.4529	94.9
Cloc	450	0.6187	**0.7369**	0.3698	0.5116	100.1
Clocorient	450	0.6312	0.7638	0.3656	0.5336	96.0
Cloc_long	1529	0.6886	0.7869	0.3314	0.6393	101.6
Cloc_80k	2500	**0.7033**	0.7762	**0.3133**	**0.6675**	103.9
LLaVA-1.5-7B (reference)	n/a	0.7910	0.5069	0.1518	0.7738	82.4

Table 4.16: Comparison against a fully-resourced reference, dd256. A bold entry marks the best of our conditions in each column; lower CHAIR is better. Steps are adapter steps at an effective batch of 32, so the rungs of the location ladder see 14,400, 48,928 and 80,000 items. The reference is evaluated for inference only, and the reference-based metrics are absent here because the detailed regime scores 256-token descriptions against one-sentence references, as Table 4.3 sets out.

4.8.2 How far a single-axis move gets

In terms of fractions of the distance from the pinned baseline to the reference, the best scenario covers 28% for CLIPScore, 26% for CHAIR_i, and 25% for recall, achieved

by Clocorient, as well as 18% for CHAIR_s, achieved by Cloc. The important thing is that it covers roughly a quarter across four metrics of two families A geometric intervention costing 450 adapter steps moves caption quality along the same direction that orders of magnitude more data moves it, rather than exploiting one metric. Cloc, which releases location alone, covers 22%, 24% and 18% of those respective distances by itself.

4.8.3 Scaling the location drive

The conditions above all run for 450 adapter steps, a budget chosen to make the factorial affordable rather than to reach the best attainable quality. Since the attribution identifies location as the axis that carries the most benefit, the follow-up is to hold that configuration fixed and train it longer. Two continuations extend Cloc on the same objective, the same pins, and the same data stream: Cloc_long to 1,529 steps (48,928 items) and Cloc_80k to 2,500 steps (80,000 items). The second resumes the first and advances the data iterator past the items the checkpointed steps already consumed, so it sees items 48,929 to 80,000 and nothing twice.

The trend is consistent on every measure but one, as Figure 4.12 shows. CLIPScore rises from 0.6187 to 0.6886 and then 0.7033, so the step from 450 to 1,529 steps is worth 16.8 standard errors, and the step to 2,500 a further 4.0: still resolved, and clearly diminishing. CHAIR_i falls from 0.3698 to 0.3314 to 0.3133, and recall rises from 0.5116 to 0.6393 to 0.6675, both with bootstrap intervals excluding the baseline at every rung. Measured against the reference, Cloc_80k covers 60% of CLIPScore, 44% of CHAIR_i and 67% of recall, against the 22%, 24% and 18% that the 450-step Cloc configuration achieved.

CHAIR_s is the exception and moves the other way: it is 0.7369 at 450 steps, 0.7869 at 1,529 and 0.7762 at 2,500, and only the first resolves against the baseline. The reason is mechanical. The model is talking more, 100.1, 101.6 and 103.9 words, naming 8.98, 9.26 and 9.67 objects per caption. CHAIR_s counts the fraction of captions that contain at least one hallucinated object, so a caption that names more objects is more exposed to it, while CHAIR_i, the fraction of named objects that are hallucinated, keeps falling. More training buys coverage and per-object precision rather than brevity, and the per-caption measure registers that as a loss.

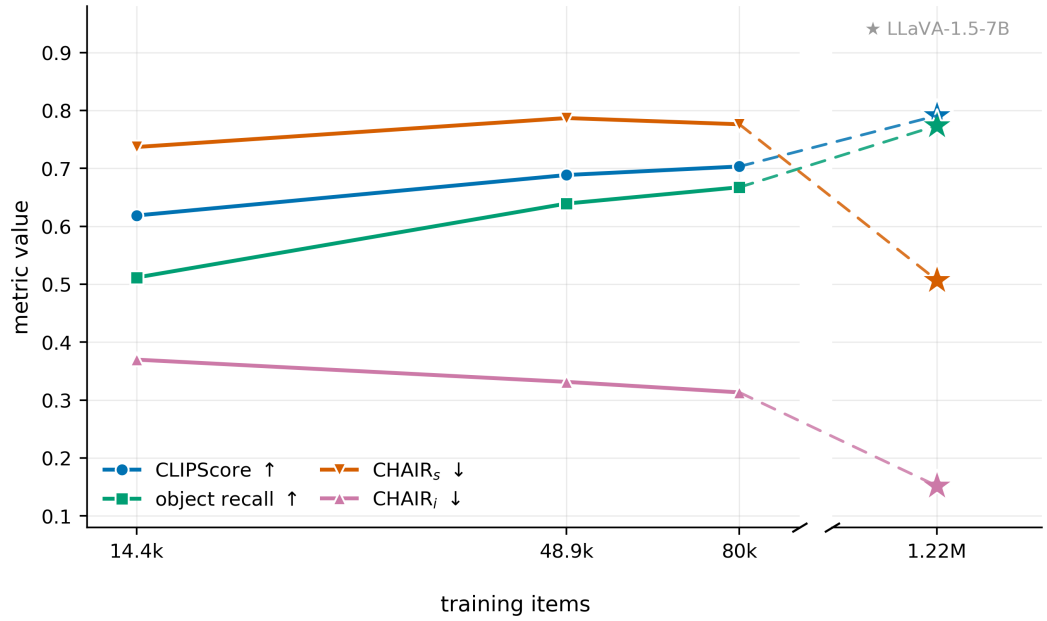


Figure 4.12: The location ladder against training volume, dd256 with n=1300. All four measures are dimensionless rates, so they share one vertical axis but are read in their own units; the arrow beside each name gives the direction of improvement. Solid segments join the three Cloc checkpoints and a dashed segment carries each metric across to its star, the LLaVA-1.5-7B reference. The horizontal axis is broken: the ladder spans 14.4k to 80k items and the reference sits at 1.22M, so a continuous axis would give one point more width than the whole ladder. The dashed segments therefore interpolate across the period the model was trained between 80k and 1.22M items. The item counts are those each location-drive run consumed on top of the two-stage baseline that every condition shares, against the reference's full training, so the two are comparable in order of magnitude rather than exactly. CLIPScore reaches 0.7033 against the reference's 0.7910 and recall 0.6675 against 0.7738, while the distance left on CHAIR$_s$ is far larger.

Figures 4.13, 4.14, and 4.15 show that none of this comes from a run that is drifting. The location gap on the training batch falls from a mean of 126 over the first hundred steps to 18.8 over the hundred ending at step 1,529 and 16.7 over the hundred ending at step 2,500, flattening rather than diverging, and the full evaluation set agrees at all three checkpoints. The autoregressive loss settles near 1.14 and stays there, so the language objective is not traded away as the gap closes. The pinned geometric statistics hold their targets throughout, and the isolation survives at the evaluation set as well: across the three rungs total variance moves from 4789.6 to 4861.3, the participation ratio from 37.32 to 37.27, and O_{16} from 0.0519 to 0.0515 against the baseline's 0.0515. Only location moves, given three and a half times the budget.

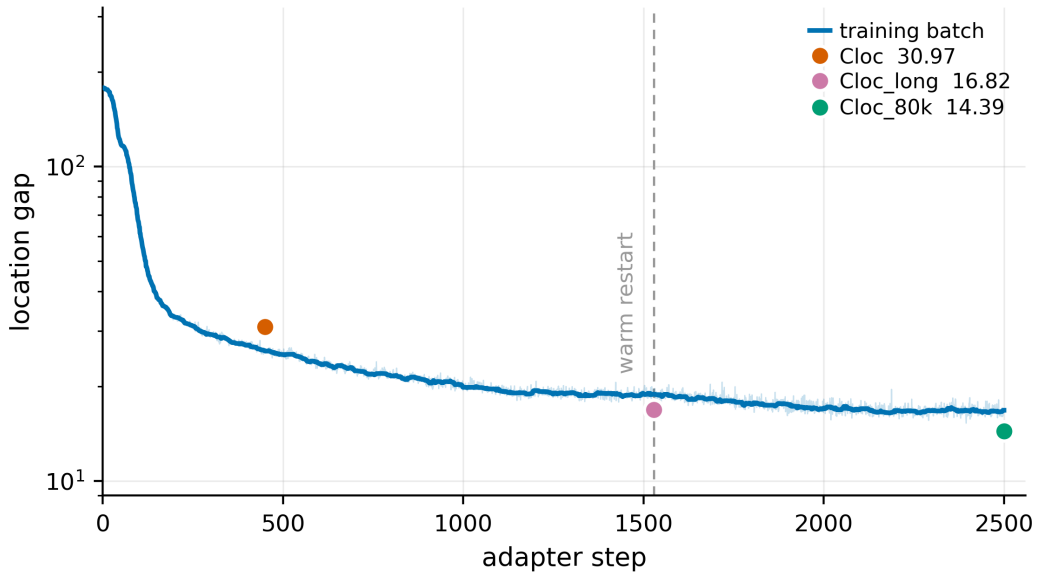


Figure 4.13: The location gap during training. The light trace is the raw loss on the training batch and the solid blue line its 25-step rolling median. The three coloured markers are the full-set G_μ at the checkpoints where it was saved and evaluated, measured over 5000 pairs rather than one batch of 32, which is why they do not sit exactly on the batch curve. The vertical line at 1529 marks the warm restart [45] where Cloc_80k resumes Cloc_long

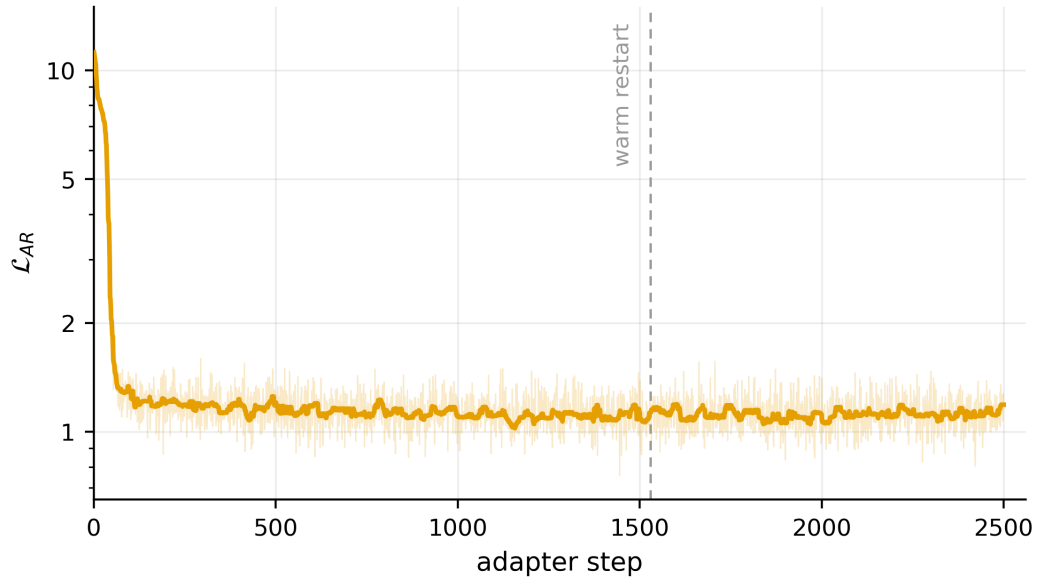


Figure 4.14: The autoregressive loss over the same 2500 steps on a logarithmic axis, per step in light and as a 25-step rolling median in bold. It falls from 11.2 at step 1 to about 1.3 within the first hundred steps, and the hundred-step means centred on steps 450, 1050, 1500 and 2450 are 1.175, 1.126, 1.123 and 1.130.

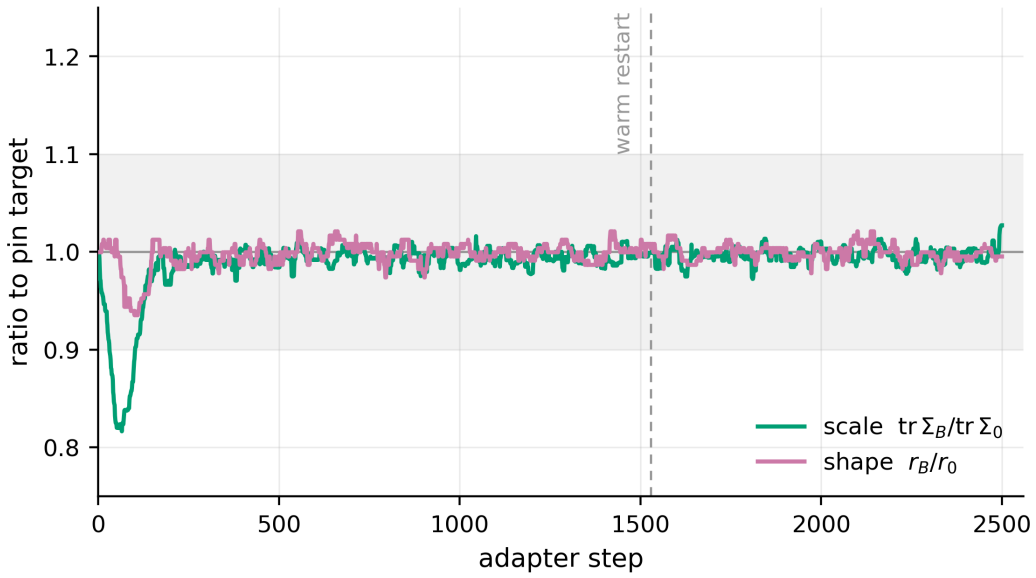


Figure 4.15: The two pinned statistics, each as a ratio to its own target, $\operatorname{tr}\Sigma_0 = 4767.3$ for scale and $r_0 = 23.31$ for shape, measured on the training batch. The shaded band is $\pm 10\%$. The opening dip is the pins taking hold over the first 150 steps; afterwards they track their targets, within 2% on average over the whole schedule and within 0.5% over the continuation.

One detail of the schedule deserves a note, because the continuation resumes a cosine that had already annealed to zero. At step 1,530 the learning rate jumps from 0 to 6.91×10^{-5}, which is 35% of the peak rate, but the trajectory absorbs it: the batch gap reads 18.6, 19.0, 19.7 and 17.7 across the four steps straddling the boundary and the autoregressive loss reads 1.18, 1.19 and 0.93, both inside the ordinary step-to-step spread.

4.8.4 The geometry of the remaining distance

However, the better comparison is geometric rather than through downstream scores. If one uses the same statistics to compute the location gap $\widehat{G}_\mu = 3.250$ for the connector output of the reference in the 1300 pairs evaluation set, it is *larger* than our baseline at 2.727 and very far from closure at 0.448. Subspace overlapping is similarly just 2 to 3 times that of chance overlap: $O_4 = 0.0026$ versus $q/d = 0.001$ and $O_{16} = 0.0086$ versus 0.0039, while our orientation-based criterion achieves 0.1404 at $q = 4$. Thus, a training size more than one order of magnitude larger was not enough to bring the reference images close to the text centroid, nor to make the two subspaces align. Deliberately shutting down the location axis indeed creates genuine headroom.

This reading is in line with the residual ratio of Equation (4.6), which is a measure of the proportion of the gap's second moment that is not due to the centroid offset. It reads $\rho_r = 0.118$ at our baseline and 0.086 for the reference, so in the decoder's input space between 88% and 91% of that second moment is the centroid offset. This is the opposite of what is reported for the normalised space of a contrastive dual

encoder, where the ratio itself is close to 0.89, so the residual rather than the centroid offset dominates (Section 2.2). The two measurements evaluate completely different spaces and are not in conflict, but the difference is the reason location is the effective lever here.

There is one caveat that should be noted about the geometric comparison. The reference cloud has a very anisotropic structure, with an effective rank of 3.42 (compared to around 37 for our cloud) and a condition number on the order of 3×10^{13}, as there are a few large-norm tokens in the mean. This results in an exaggerated value of G_μ. The clean confirmation would be to test in the reference's own space, by shifting its connector output towards its own text centroid and re-measuring; that test is left to future work.

4.8.5 Qualitative examples along the location ladder

The averages of Table 4.16 hide what the extra training changes in a single caption. The three images below are the cases where Cloc_80k gains most over the pinned baseline, and each is shown for the whole ladder together with the reference. The pattern is the same in all three: the baseline describes a scene that is not there, each rung removes hallucinated objects, and the last rung reaches or passes the reference.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a **man** and a **woman** sitting on a **couch**, watching a **television**. The **man** is holding a **remote** control in his hand, and the **woman** is holding a **cup** of coffee. They are both dressed in casual clothing, and the room they are in appears to be a living room. The **television** is placed on a stand, and there is a small **table** with a **cup** of coffee on it. The room also has a lamp and a few other items, such as a **clock** and a **vase**. The scene is cozy and comfortable, with the couple enjoying their time together.	0.428	13	0
Cloc (450)	The image shows a couple of **people** sitting on a bed in a room. The **man** is sitting on the bed, and the **woman** is sitting next to him. They are both holding a **baby**, with the **baby** sitting on the **man**'s lap. The room is decorated with a few items, including a teddy bear and a small **table**. The bed is made up with a blanket and pillows, and there is a lamp on the **table**. The room has a cozy and comfortable atmosphere.	0.751	8	2
Cloc_long (1529)	The image features a couple of stuffed animals, a teddy bear and a **dog**, lying on a bed together. The teddy bear is positioned on the left side of the bed, while the **dog** is on the right side. The bed is covered with a white blanket, and the stuffed animals are placed on top of it. There are two **chairs** in the room, one on the left side and the other on the right side of the bed. A **clock** is also visible on the wall, adding to the cozy atmosphere of the scene.	0.839	4	2
Cloc_80k (2500)	The image features two stuffed animals, a brown teddy bear and a pink bear, lying on a bed together. The teddy bear is positioned on the left side of the bed, while the pink bear is on the right side. The bed is covered with a white blanket, and the stuffed animals are resting comfortably on it. There are two pillows on the bed, one on the left side and the other on the right side. The room is dimly lit, creating a cozy atmosphere for the stuffed animals.	0.867	0	3
LLaVA-1.5-7B	The image features a cozy bed with a white comforter and a pillow. On the bed, there are three stuffed animals, including two teddy bears and a stuffed **dog**. The teddy bears are positioned close to each other, while the stuffed **dog** is situated further away. The arrangement of the stuffed animals creates a warm and inviting atmosphere in the bedroom.	0.878	2	2

Figure 4.16: Teddy bears on a bed. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bear, bed, teddy bear.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a large, comfortable living room with a **couch** and a **chair** placed in front of a fireplace. The room is well-lit, with a lamp on the coffee **table** and a lamp on the side **table**. There is a **television** mounted on the wall, and a book is placed on the coffee **table**. The room is spacious and inviting, with a rug on the floor and a rug on the **couch**. A **woman** is seated on the **couch**, reading a book, while a **man** is standing in the room. There is a cat in the room as well, adding to the cozy atmosphere. The room is decorated with various items, including a **vase** on the coffee **table**, a **clock** on the wall, and a plant on the side **table**.	0.559	14	2
Cloc (450)	The image features a living room with a large, comfortable **couch** and a **television** set. The **couch** is situated in the center of the room, and it is surrounded by various items, including a laptop, a book, and a **remote** control. There are also two pillows on the **couch**, one on the left side and the other on the right side. The room is spacious and well-lit, with a window in the background that provides natural light. The room also has a rug on the floor, adding a touch of warmth and comfort to the space.	0.610	5	2
Cloc_long (1529)	The image features a black and white cat sitting on a bed, with a book on its lap. The cat is looking at the camera, capturing the viewer's attention. The bed is situated in a room with a window, and there are several books scattered around the room. In addition to the cat, there are two **chairs** in the room, one on the left side and another on the right side. A **clock** is also visible on the wall, adding to the cozy atmosphere of the room.	0.718	2	3
Cloc_80k (2500)	The image features a black and white cat sitting on a bed, surrounded by a variety of books. The cat is lying on the bed, with its head resting on a pillow. The books are scattered around the bed, some placed on the bed itself, while others are on the floor. There are at least 12 books visible in the scene, with some placed closer to the cat and others further away. The cat appears to be relaxed and comfortable in its cozy environment.	0.697	0	3
LLaVA-1.5-7B	The image features a black cat lying on a bed, which is covered with a plaid blanket. The cat is positioned in the middle of the bed, occupying a significant portion of the space. The room also contains a bookshelf filled with various books, some of which are placed on the bed as well. There are at least 13 books visible in the scene, with some books located near the cat and others scattered throughout the room. The overall atmosphere of the room is cozy and inviting.	0.789	0	3

Figure 4.17: A cat asleep on a bed beside a bookshelf. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bed, book, cat, laptop, teddy bear.



Condition	Generated caption	CLIP	H	R
all-pinned	The image features a kitchen with a large wooden **table** in the center. The **table** is surrounded by **chairs**, and there are several **people** sitting on the **chairs**. The **table** is covered with a white tablecloth, and there are several **cups** and plates on the **table**. In the background, there are several cabinets and a **refrigerator**, which can be seen behind the **people**. The kitchen appears to be well-equipped and functional, providing a comfortable and inviting space for gathering and dining.	0.385	10	0
Cloc (450)	The image shows a bathroom with a sink and a toilet, both located in the same room. The sink is located on the left side of the room, while the toilet is on the right side. The sink is equipped with a faucet, and there are two towels hanging on the wall behind it. The toilet is a white toilet with a **seat** and a lid, and it is positioned in front of the sink. There are two mirrors in the room, one on the left wall and the other on the right wall. The mirror on the left wall is placed above the sink, while the mirror on the right wall is positioned above the toilet. There is also a window in the room, which is located on the right side of the bathroom.	0.788	0	2
Cloc_long (1529)	The image features a bathroom with a white sink and a toilet. The sink is located on the left side of the room, while the toilet is on the right side. The sink is equipped with a faucet, and there is a towel hanging on the wall above it. The bathroom is spacious and well-lit, with a window on the right side of the room. There are two mirrors in the bathroom, one above the sink and another one on the wall behind the toilet. A bottle can be seen on the counter next to the sink, and a **cup** is placed on the counter as well.	0.810	1	3
Cloc_80k (2500)	The image features a bathroom with a white sink and a toilet. The sink is located on the left side of the room, while the toilet is on the right side. The sink is equipped with a faucet, and there is a towel hanging on the wall above it. The bathroom is spacious and well-lit, with a window on the right side of the room, allowing natural light to enter. There is also a mirror on the wall, providing a reflection of the room's contents. A bottle can be seen on the countertop, possibly containing soap or other bathroom essentials.	0.808	0	3
LLaVA-1.5-7B	The image depicts a small bathroom with a white toilet situated next to a sink. The bathroom is equipped with a shower and a bathtub, which is located near the toilet. There are two bottles of cleaning products placed on the counter, one near the sink and the other closer to the toilet. Additionally, there are two **cups**, one near the sink and the other on the counter. The bathroom appears to be well-maintained and organized.	0.768	1	3

Figure 4.18: A bathroom with a sink and a toilet. Conditions are listed in order of training budget; the reference sits below the rule. CLIP is the reference-free CLIPScore of that caption against the image; objects named but absent are marked in **red**, H counts them and R counts the annotated objects recovered. The annotated objects are bottle, sink, toilet.

Chapter 5

Conclusions

This thesis asked which property of the modality gap a frozen language decoder is sensitive to when it has to describe an image. The gap is normally reported as a single centroid distance, measured inside a contrastive dual encoder and corrected by one transformation that moves several properties at once. Answering that required treating it as a composite object and building conditions in which one property moves while the others are held fixed and verified to have held.

5.1 What was established

The four axes behave differently, and the differences are large enough to be unambiguous. Closing the centroid distance while scale, shape and orientation stay pinned raises reference-free image-text agreement by 0.0526 CLIPScore, worth 22.9 standard errors, and improves both hallucination rates and object recall at once. The gain is not obtained by saying less: the treated model names more objects per caption, 8.98 against 8.69, in captions 5.2 words longer, and is correct about a larger fraction of them. Compressing total variance produces the exact mirror of that result on every metric. Redistributing variance over fewer directions changes caption quality by nothing measurable. Orientation, the axis a contrastive objective controls most directly and the easiest to move, returns roughly a third of what location returns and shifts no hallucination measure past its interval.

The reason those statements can be made is methodological. A contrastive objective swept over its weight moves all four axes monotonically and together, so no correlation across such a sweep can assign an effect to any one of them. The factorial design separates them, and the presence of both a mirror and a null condition is what makes the attribution defensible: a perturbation that helps, one that hurts and one that does neither cannot all be generic disturbance of the decoder's input. Four ranking measures from two sources of ground truth order the eight conditions identically, so the result does not depend on the metric family.

Two findings constrain how the location effect should be read. It is anchor-agnostic, since replacing the embedding-table target with a frozen CLIP text tower leaves it unchanged, which points to translation into a region the decoder reads well

rather than to alignment with meaning. And it is visible only when the objective acts on the same pooled vector the decoder consumes; written against the class token, the same objective at the same weight moves the centroid roughly six times less. The gap is also partly self-inflicted: contrastive pre-training sets orientation and scale, and instruction tuning then moves the visual centroid 38% further from the text manifold, having no alignment term to prevent it.

The axis keeps paying when the configuration is trained longer. Extending the drive from 450 to 2,500 adapter steps, with both pins holding, carries CLIPScore from 0.6187 to 0.7033 and object recall from 0.5116 to 0.6675, which is 60% and 67% of the distance from the pinned baseline to a fully-resourced reference trained on roughly fifteen times more data. The improvement is monotone and clearly diminishing, and the training trace shows no collapse.

5.2 Limitations

The models are instruments, not competitive systems: 450 optimizer steps, about 9.1% of one epoch, on a single 11 GB GPU. Every claim is a contrast between conditions sharing that budget exactly, but the conditions under which the attribution holds are correspondingly narrow.

5.3 Future directions

Two things would settle what this budget left open, and both extend runs already reported. The first is to train every condition for a full epoch. The factorial was fixed at 450 steps, about 9.1% of one epoch of LLaVA-Instruct-150K, so that six conditions fitted the available hardware. Only the location drive was carried further, and it kept improving: CLIPScore rose from 0.6187 to 0.7033 between 450 and 2,500 steps, the latter still only half an epoch, monotone and clearly diminishing but not exhausted. A condition harmful at 450 steps might be slow to recover rather than harmful, and an axis inert over a short run is not thereby inert over a long one.

The second is to repeat each condition across seeds. Every run here used a single seed, so the reported standard errors describe variation across evaluation images rather than across runs. The headline effects are large enough that run-to-run noise is unlikely to overturn them.

Appendix A

Implementations

This appendix gives the code for the three components the thesis depends on: the connector, the geometry objective of Section 3.2, and the measurement of the orientation axis of Section 3.3. The listings are the working implementations with logging and diagnostics removed, not pseudocode.

A.1 The connector

The connector is a two-layer perceptron applied token-wise, so a batch of 257 projected tokens is produced by one matrix product per layer and no reshaping is required.

Listing A.1: The connector, a two-layer MLP with a GELU non-linearity, mapping CLIP token features into the input embedding space of the decoder.

```
class MLP2xGELU(nn.Module):
    def __init__(self, in_dim=1024, hidden_dim=4096, out_dim=4096):
        super().__init__()
        self.fc1 = nn.Linear(in_dim, hidden_dim)
        self.act = nn.GELU()
        self.fc2 = nn.Linear(hidden_dim, out_dim)

    def forward(self, x):
        # (B, 257, 1024) -> (B, 257, 4096); acts token-wise
        return self.fc2(self.act(self.fc1(x)))
```

A.2 The geometry objective

Listing A.2 computes the three geometric terms on one batch. Three details in it are load-bearing and are easy to lose when reimplementing.

The vector the terms act on is the mean over the 257 projected tokens, not the projected `[CLS]` token. This is the same vector the decoder consumes and the metrics report, and Section 4.5.4 measures what acting on the wrong one costs. The terms are evaluated in `float32` even when the forward pass runs in `bfloat16`, because a squared centroid offset in 4096 dimensions is poorly conditioned in half precision; the gradient still flows back into the connector. And the two pin setpoints are captured

once, at the first geometry step, and frozen from then on. A setpoint recomputed each step would follow the cloud it is meant to hold, which is how the pin fails silently.

The participation ratio is computed from the $B \times B$ Gram matrix of the centred batch rather than from the ambient covariance. The two share their non-zero eigenvalues, so the 4096×4096 covariance is never formed, which is what makes the shape axis controllable on a batch at all.

Listing A.2: The three geometric terms. `mu_y` is the frozen text centroid and `trace_x` the image-cloud trace, both precomputed; `state` carries the frozen pin setpoints.

```
def geometry_terms(vis, projector, mu_y, trace_x, state):
    # control == measurement: the pooled vector, not the projected [CLS] token
    z = projector(vis).mean(dim=1)                  # (B, 4096)
    zf = z.float()                                  # fp32 for a conditioned term
    zbar = zf.mean(dim=0)                           # (4096,) batch centroid

    # LOCATION: squared centroid offset, scaled by the cloud trace so that
    # lambda_d is comparable across conditions.
    l_dist = ((zbar - mu_y) ** 2).sum() / trace_x

    # SCALE pin: batch spread held at its first-step value, so the optimiser
    # must translate the cloud instead of shrinking it onto mu_y.
    btrace = ((zf - zbar) ** 2).sum(dim=1).mean()
    if state.btrace0 is None:
        state.btrace0 = float(btrace.detach())  # frozen from here on
    l_scale = (btrace / state.btrace0 - 1.0) ** 2

    # SHAPE pin: participation ratio from the B x B Gram matrix of the centred
    # batch, whose non-zero eigenvalues are the covariance's. The 4096 x 4096
    # ambient covariance is never formed.
    zc = zf - zbar
    G = zc @ zc.t()                                 # (B, B)
    pr = torch.diagonal(G).sum() ** 2 / (G * G).sum().clamp_min(1e-12)
    if state.effrank0 is None:
        state.effrank0 = float(pr.detach())
    l_rank = (pr / state.effrank0 - 1.0) ** 2

    return l_dist, l_scale, l_rank
```

The geometric terms and the language objective are combined convexly, so that raising λ_d takes budget from the autoregressive loss rather than adding to it. The two legs are backpropagated separately: the language loss is micro-batched over the

gradient-accumulation window, and the geometry leg runs once per optimiser step on its own batch of 64 images.

Listing A.3: Combining the two legs. The weight on the language objective is $w_{\mathrm{AR}} = 1 - \lambda_d$, so the geometry term is never free; when an orientation drive is added, the split is three-way, $w_{\mathrm{AR}} = 1 - \lambda_d - \lambda_o$.

```
w_ar = 1.0 - lambda_d
for micro in ar_window:                                  # language leg,
    accumulated
    (w_ar * vlm(**micro).loss / accum).backward()

l_dist, l_scale, l_rank = geometry_terms(...)     # geometry leg, once
    per step
(lambda_d * l_dist
 + lambda_s * l_scale
 + lambda_r * l_rank).backward()
optimizer.step()
```

A.3 Measuring the orientation axis

The subspace overlap of Equation (3.3) is measure offline on saved embeddings rather than during training, in `float64`, because the leading eigenvectors of a near-singular covariance are sensitive to precision.

Listing A.4: Subspace overlap O_q between the top-q principal subspaces of the two clouds. The chance level for a random q-dimensional subspace is q/d.

```
def subspace_overlap(X, Y, q):
    X, Y = X.astype(np.float64), Y.astype(np.float64)
    _, Ux = eigh_descending(np.cov(X, rowvar=False))
    _, Uy = eigh_descending(np.cov(Y, rowvar=False))
    M = Ux[:, :q].T @ Uy[:, :q]
    return float(np.linalg.norm(M, "fro") ** 2 / q)
```

Bibliography

[1] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. "Visual Instruction Tuning". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2023 (cit. on pp. 1, 5, 7, 8, 18, 27).

[2] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models". In: *International Conference on Machine Learning (ICML)*. 2023 (cit. on pp. 1, 5, 7, 18).

[3] Alec Radford et al. "Learning Transferable Visual Models From Natural Language Supervision". In: *International Conference on Machine Learning (ICML)*. 2021 (cit. on pp. 1, 5–7, 17, 18).

[4] Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. "Mind the Gap: Understanding the Modality Gap in Multi-modal Contrastive Representation Learning". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2022 (cit. on pp. 1, 8–10).

[5] Yuhui Zhang, Elaine Sui, and Serena Yeung-Levy. "Connect, Collapse, Corrupt: Learning Cross-Modal Tasks with Uni-Modal Data". In: *arXiv preprint arXiv:2401.08567* (2024) (cit. on pp. 1, 11).

[6] Xiaomin Yu et al. *Modality Gap-Driven Subspace Alignment Training Paradigm For Multimodal Large Language Models*. 2026. arXiv: 2602.07026 [cs.CV]. URL: https://arxiv.org/abs/2602.07026 (cit. on pp. 1, 11–13, 18).

[7] Xiaomin Yu et al. *Anisotropic Modality Align*. 2026. arXiv: 2605.07825 [cs.MM]. URL: https://arxiv.org/abs/2605.07825 (cit. on pp. 1, 11–13, 18).

[8] Chao Jia et al. "Scaling Up Visual and Vision-Language Representation Learning with Noisy Text Supervision". In: *International Conference on Machine Learning (ICML)*. 2021. arXiv: 2102.05918 (cit. on p. 5).

[9] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. "Improved Baselines with Visual Instruction Tuning". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024 (cit. on pp. 5, 7, 18, 50).

[10] Jean-Baptiste Alayrac et al. "Flamingo: A Visual Language Model for Few-Shot Learning". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2022. arXiv: 2204.14198 (cit. on p. 5).

[11] Chameleon Team. "Chameleon: Mixed-Modal Early-Fusion Foundation Models". In: *arXiv preprint arXiv:2405.09818* (2024) (cit. on p. 5).

[12] Wonjae Kim, Bokyung Son, and Ildoo Kim. "ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision". In: *International Conference on Machine Learning (ICML)*. 2021 (cit. on p. 5).

[13] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. "UNITER: Universal Image-Text Representation Learning". In: *European Conference on Computer Vision (ECCV)*. 2020 (cit. on p. 5).

[14] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. "A Survey on Multimodal Large Language Models". In: *National Science Review* 11.12 (2024), nwae403. DOI: 10.1093/nsr/nwae403 (cit. on pp. 6, 7).

[15] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation Learning with Contrastive Predictive Coding". In: *arXiv preprint arXiv:1807.03748* (2018) (cit. on pp. 6, 18).

[16] Tongzhou Wang and Phillip Isola. "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere". In: *International Conference on Machine Learning (ICML)*. 2020. arXiv: 2005.10242 (cit. on pp. 7, 11).

[17] Boya Wu et al. *Bunny-v1.1 training data*. Hugging Face dataset BoyaWu10/Bunny-v1_1-data 2024 (cit. on pp. 7, 27).

[18] Hugo Touvron, Louis Martin, Kevin Stone, et al. "Llama 2: Open Foundation and Fine-Tuned Chat Models". In: *arXiv preprint arXiv:2307.09288* (2023) (cit. on pp. 7, 17).

[19] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. "LoRA: Low-Rank Adaptation of Large Language Models". In: *International Conference on Learning Representations (ICLR)*. 2022 (cit. on pp. 7, 16, 18).

[20] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. "QLoRA: Efficient Finetuning of Quantized LLMs". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2023 (cit. on pp. 7, 17).

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft COCO: Common Objects in Context". In: *European Conference on Computer Vision (ECCV)*. 2014 (cit. on pp. 9, 14, 27).

[22] Peiyang Shi, Michael C. Welle, Mårten Björkman, and Danica Kragic. "Towards Understanding the Modality Gap in CLIP". In: *ICLR Workshop on Multimodal Representation Learning: Perks and Pitfalls*. 2023 (cit. on p. 10).

[23] Simon Schrodi, David T. Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. "Two Effects, One Trigger: On the Modality Gap, Object Bias, and Information Imbalance in Contrastive Vision-Language Representation Learning". In: *International Conference on Learning Representations (ICLR)*. 2025. arXiv: 2404.07983 (cit. on p. 10).

[24] Abrar Fahim, Alex Murphy, and Alona Fyshe. *It's Not a Modality Gap: Characterizing and Addressing the Contrastive Gap*. 2024. arXiv: 2405.18570 [cs.LG] (cit. on p. 10).

[25] Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. "Representation Degeneration Problem in Training Natural Language Generation Models". In: *International Conference on Learning Representations (ICLR)*. 2019. arXiv: 1907.12009 (cit. on p. 11).

[26] Kawin Ethayarajh. "How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings". In: *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2019, pp. 55–65. DOI: 10.18653/v1/D19-1006 (cit. on p. 11).

[27] Jiaqi Mu and Pramod Viswanath. "All-but-the-Top: Simple and Effective Postprocessing for Word Representations". In: *International Conference on Learning Representations (ICLR)*. 2018. arXiv: 1702.01417 (cit. on p. 11).

[28] Sedigheh Eslami and Gerard de Melo. "Mitigate the Gap: Improving Cross-Modal Alignment in CLIP". In: *International Conference on Learning Representations (ICLR)*. 2025. arXiv: 2406.17639 (cit. on p. 12).

[29] Xiaomin Yu, Wenjie Zhang, Ziyue Qiao, Chengwei Qin, and Hui Xiong. *Text-Only Data Synthesis for Vision Language Model Training*. Introduces the Unicorn-1.2M and Unicorn-471K-Instruction datasets. 2025. arXiv: 2503.22655 [cs.AI]. URL: https://arxiv.org/abs/2503.22655 (cit. on p. 12).

[30] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. "Show and Tell: A Neural Image Caption Generator". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. arXiv: 1411.4555 (cit. on p. 14).

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "BLEU: A Method for Automatic Evaluation of Machine Translation". In: *Annual Meeting of the Association for Computational Linguistics (ACL)*. 2002 (cit. on pp. 14, 28).

[32] Satanjeev Banerjee and Alon Lavie. "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments". In: *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. 2005 (cit. on pp. 14, 28).

[33] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. "CIDEr: Consensus-Based Image Description Evaluation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015 (cit. on pp. 14, 28).

[34] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. "SPICE: Semantic Propositional Image Caption Evaluation". In: *European Conference on Computer Vision (ECCV)*. 2016 (cit. on pp. 14, 28).

[35] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. "CLIPScore: A Reference-free Evaluation Metric for Image Captioning". In: *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2021 (cit. on pp. 14, 27).

[36] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. "Object Hallucination in Image Captioning". In: *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2018 (cit. on pp. 14, 28).

[37] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. "Evaluating Object Hallucination in Large Vision-Language Models". In: *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2023. arXiv: 2305.10355 (cit. on p. 15).

[38] Alexey Dosovitskiy et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations (ICLR)*. 2021. arXiv: 2010.11929 (cit. on p. 17).

[39] Dan Hendrycks and Kevin Gimpel. *Gaussian Error Linear Units (GELUs)*. 2016. arXiv: 1606.08415 [cs.LG] (cit. on p. 17).

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention Is All You Need". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2017 (cit. on p. 17).

[41] Olivier Roy and Martin Vetterli. "The Effective Rank: A Measure of Effective Dimensionality". In: *European Signal Processing Conference (EUSIPCO)*. 2007, pp. 606–610 (cit. on p. 20).

[42] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019 (cit. on p. 26).

[43] Ilya Loshchilov and Frank Hutter. "Decoupled Weight Decay Regularization". In: *International Conference on Learning Representations (ICLR)*. 2019. arXiv: 1711.05101 (cit. on p. 26).

[44] Wei-Lin Chiang et al. *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. https://lmsys.org/blog/2023-03-30-vicuna/. 2023 (cit. on p. 50).

[45] Ilya Loshchilov and Frank Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts". In: *International Conference on Learning Representations (ICLR)*. 2017 (cit. on p. 53).