Uncovering the Inner Workings of STEGO for Safe Unsupervised Semantic Segmentation

Alexander Koenig, Maximilian Schambach, Johannes Otterbach
Merantix Momentum, Berlin, Germany
{firstname.lastname}@merantix.com

Full Paper

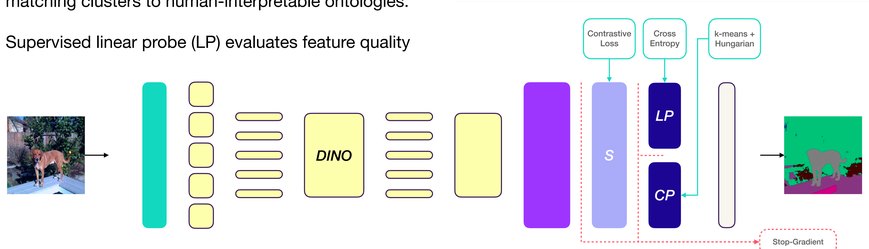
JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

1. Introduction

- Problem: labeled data is scarce, but unlabeled data is abundant
- Self-supervised learning recently demonstrated impressive results on unlabeled datasets
- For example, STEGO [1] is an algorithm for unsupervised semantic segmentation
- To apply the STEGO algorithm safely in real-world settings, it's crucial to understand its working mechanisms

2. STEGO Architecture & Training Strategy

- STEGO builds on the DINO [2] pre-trained Vision Transformer, see **Figure 1**
- STEGO's segmentation head S non-linearly projects DINO features into lower-dimensional space, "distilling" DINO feature correspondences
- S is trained using a self-supervised contrastive loss that involves 6 hyper-parameters
- Unsupervised cluster probe (CP) maps STEGO features to ontologies using k-Means and Hungarian Matching algorithm. Labels are only needed for matching clusters to human-interpretable ontologies.
- Supervised linear probe (LP) evaluates feature quality



Operation	Read Image	Patch Image	Embed Tokens	...	Output Tokens	Reshape Tokens	Upsample Tokens	Segment. Head	Linear Probe / Cluster Probe	CRF Refine
# Outputs	1	40^2	40^2		40^2	1	1	1	1	1
Channels	3	3	D_{ViT}		D_{ViT}	D_{ViT}	D_{ViT}	D_{STEGO}	N_c	N_c

Figure 1: Architecture of STEGO validation pipeline.

3. Re-establishing Baselines

- **Table 1** shows results of our reproducibility study
- Evaluating pre-trained models in "STEGO (ours)" ✓ gives same results as "STEGO (theirs)"
- However, training unsupervised cluster probe on raw DINO features, under "DINO (ours)", performs better † than suggested in STEGO paper
- Also, training supervised linear probe on raw DINO features, under "DINO (ours)", performs approximately equal † or greater ‡ than on segm. head features

Method	Cocostuff				Cityscapes				Potsdam			
	Unsupervised Cluster probe		Supervised Linear probe		Unsupervised Cluster probe		Supervised Linear probe		Unsupervised Cluster probe		Supervised Linear probe	
	Acc	mIoU	Acc	mIoU	Acc	mIoU	Acc	mIoU	Acc	mIoU	Acc	mIoU
STEGO (theirs)	56.9	28.2	76.1	41.0	73.2	21.0	-	-	77.0	-	-	-
STEGO (ours)	✓56.9	✓28.2	✓76.1	✓41.1	✓73.2	✓21.0	89.6	28.0	✓77.0	62.6	85.9	74.8
DINO (theirs)	30.5	9.6	66.8	29.4	-	-	-	-	-	-	-	-
DINO (ours)	†42.4	†13.0	†75.8	‡44.4	52.6	15.2	‡91.3	‡34.9	71.3	54.3	84.5	72.8

Table 1: Validation results of reproducibility study.

4. Disentangling STEGO's Working Principles

- Segmentation head S reduces dimensionality from $D_{\text{ViT}} = 768$ to $D_{\text{STEGO}} = 90$ for Cocostuff. The k-Means algorithm converges better in lower dimensions due to the curse of dimensionality.
- **Hypothesis**: STEGO is a semantics-preserving dimensionality reduction technique specifically suited for k-Means style clustering algorithms
- Steep cumulative explained curve in **Figure 2** indicates great potential for dimensionality reduction techniques on ViT features

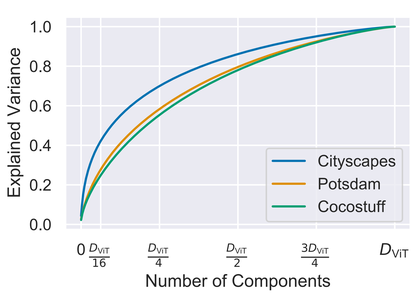


Figure 2: Cumulative explained variance of principal components of DINO features.

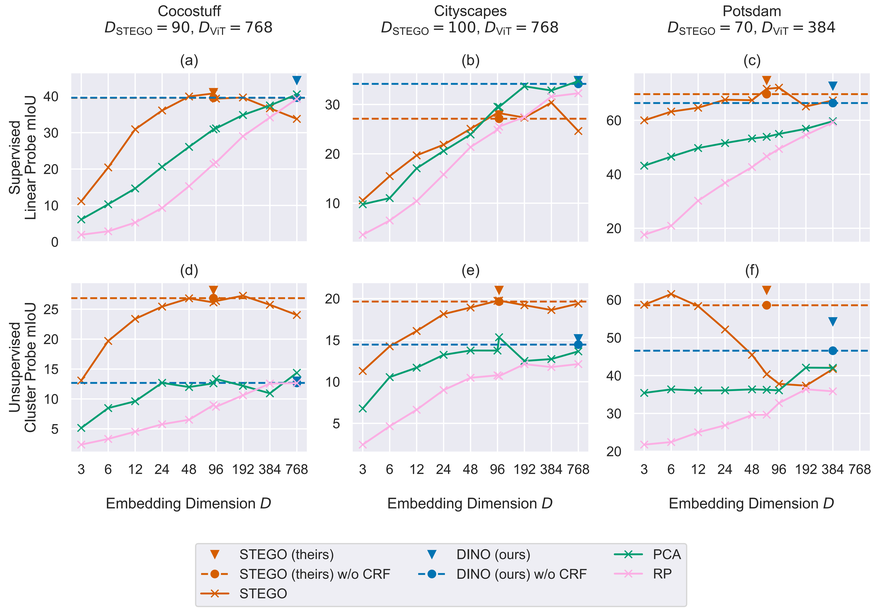


Figure 3: Validation mIoU of CP and LP after dimensionality reduction across projection dimensions.

- **Figure 3** supervised linear probe: STEGO is a dimensionality reduction technique, since it can robustly down-project embedding dimension by a factor of ~8 with little loss in LP performance
- **Figure 3** unsupervised cluster probe: STEGO outperforms PCA and Random Projection (RP) dim. reduction baselines
  - Reason 1: non-linear projection forms more distinct clusters (see performance at unreduced dimension D_{ViT})
  - Reason 2: while there is less information content for lower dimensions, lower dimensions favor k-Means convergence

5. Conclusion

- Vanilla DINO is highly performant
- STEGO is a semantics-preserving dimensionality reduction technique, outperforming PCA and RP baselines
- Future Work 1: STEGO's segmentation head can adapt to new data distribution after training using a contrastive loss. Can the segmentation head be trained using the simpler DINO loss, too?
- Future Work 2: What is the impact of using clustering algorithms specifically designed for high-dimensional embedding spaces?

References

[1] Hamilton et al. "Unsupervised semantic segmentation by distilling feature correspondences", ICLR 22.

[2] Caron et al. "Emerging properties in self-supervised vision transformers", ICCV 21.