DSLR : Dynamic to Static LiDAR Scan Reconstruction Using Adversarially Trained Autoencoder

P. Kumar*[1], S. Sahoo*[1], V. Shah[1], V. Kondameedi[1], A. Jain[1], A. Verma[1], C. Bhattacharyya[1], V Vinay[23]
[1]Indian Institute of Science, Bangalore, India [2]AMIDC Pvt Ltd, Bangalore, India [3]Chennai Mathematical Institute, Chennai, India

Code, data, video at dslrproject.github.io/dslr

The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)

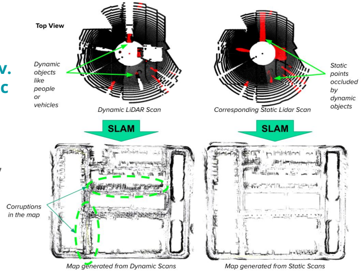
SLAM In Dynamic Env Is An Open Problem

Most **SOTA SLAM algorithms**

→ **assume a static env.**
→ thus, **fail in dynamic env.**

For LiDAR-based SLAM, we observe that

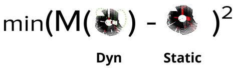
SLAM$_{\text{Dynamic}}$ << **SLAM**$_{\text{Static}}$



We address this problem by translating dynamic LiDAR scans to corresponding static LiDAR scans.

Dynamic to Static Translation (DST) for LiDAR

Learn **mapping M, from dynamic to corresponding static scans** such that the reconstruction loss is minimized



3 major goals:
→ Accurately **reconstructing the static structures** like walls or poles.
→ **Inpainting the occluded regions** with static background.
→ **Without using segmentation** information.

Challenges

→ Existing DST works for images **require segmentation** information.
→ Point cloud scan completion based methods do not work for **360° LiDAR scans**.
→ Existing LiDAR reconstruction methods fail to produce **SLAM-worthy reconstructions**.
→ **No datasets** with dynamic-corresponding static pairs exists.

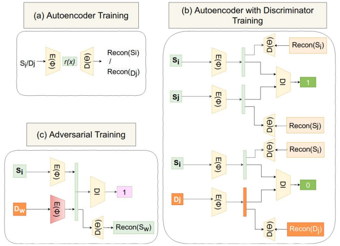
We attempt to address these challenges in our proposed approaches.

DSLR

a) Autoencoder for LiDAR scans.

b) Pair Discriminator
→ Discr. *(static, static) vs (static, dynamic)*
→ Tries to **focus on occluded regions for discrimination**, w/o segmentation.
→ **Trained on latent**, not pixel space.
→ *Dual loss* helps train using discriminative and generative features on latent space.

c) Adversarial Model *combine module (a) and (b) in an adversarial setting.*
→ Uses **adversarial label for (static, dynamic) pair (i.e. 1)**, and adjust only encoder (in red), we *map a dynamic scan to its corresponding static scan on the latent space*.

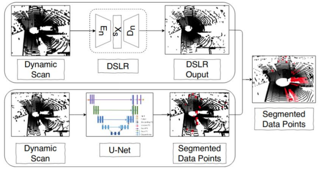


→ Also, **fine-tune encoder using reconstruction loss** b/w the input dynamic(D_W) and its corresponding static scan (S_W).

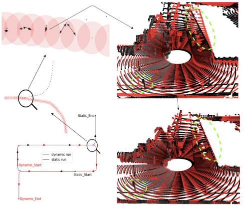
DSLR-Seg : If Segmentation Is Available

Given a dynamic frame:

→ DSLR gives static reconstruction (top)
→ U-Net finds dynamic points (bottom)
→ Replace **dynamic points with reconstructed static points** for final output (right).



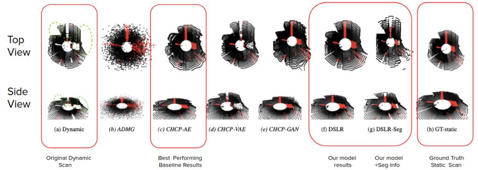
Dataset Generation



Build **first-of-its-kind real-world** paired datasets.

Dynamic scans in red, static scans in black.
→ (a), (b) show **overlapping random dynamic & static run**.
→ (c) corresponding pairs shown with arrows
→ (d) corresponding LiDAR scan pairs with *some alignment mismatch*
→ (e) No mismatch after applying **relative pose transformation**.

Experiments: DST Results

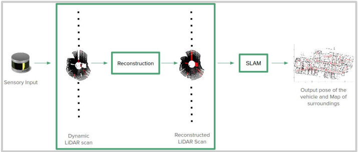


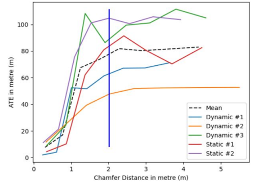
→ Our methods (second last block), clearly reconstruct better.
→ Reconstruction from DSLR **fills occluded regions, maintains walls and edges** of surrounding scenes clearly.

→ Our approach **outperform existing baselines** on all LiDAR reconstruction metrics for simulated and real world datasets.

Model	Uses Seg	Carla-64			ARD-16		KITTI-64
		EMD	Chamfer	LQI	EMD	Chamfer	LQI
AtlasNet	No	5681.85	5109.98	-	1464.61	176.46	-
ADMG	No	397.94	6.23	7.049	309.64	1.62	2.911
CHCP-VAE	No	343.98	9.58	4.080	88.94	0.67	1.128
CHCP-GAN	No	329.38	8.19	3.519	65.24	0.38	1.133
CHCP-AE	No	253.91	4.05	3.720	65.40	0.31	1.738
WCZC	Yes	$2.73*10^6$	478.12	-	-	-	-
EmptyCities	Yes	640.97	29.39	-	-	-	-
DSLR (Ours)	No	**232.51**	**1.00**	**3.350**	**57.75**	**0.20**	1.120
DSLR++ (Ours)	Yes	205.48	0.49	-	-	-	
DSLR-Seg (Ours)	Yes	**150.90**	**0.02**	-	-	-	
DSLR-UDA(Ours)	Yes	-	-	-	-	-	**1.119**

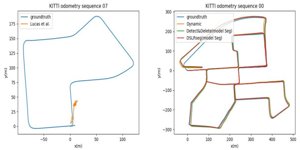
Experiments: SLAM





→ Dynamic scans, translated to its corresponding static, are consumed by SLAM giving *pose and map output.*

→ We define, **SLAM Recon. Threshold** (SRT): maximum LiDAR reconstruction error tolerable by SLAM.



→ SOTA DST methods fail on simple runs of real world datasets e.g. KITTI.
→ Our approaches **works well even on complex runs** with improved SLAM performance