# Self-Adapting Large Visual-Language Models to Edge Devices across Visual Modalities

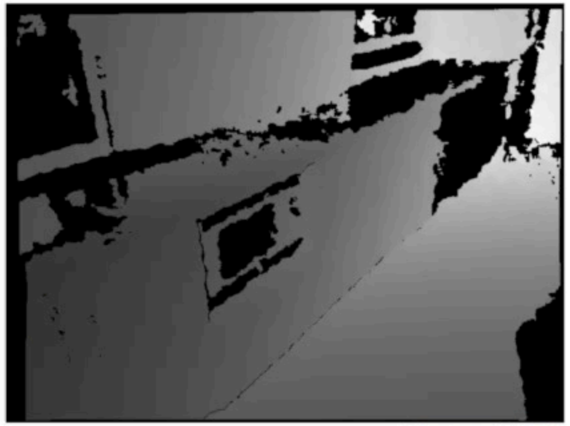
*https://github.com/ramdrop/edgevl*

Kaiwen Cai[1], Zhekai Duan[3], Gaowen Liu[2], Charles Fleming[2], Chris Xiaoxuan Lu[3]
[1]University of Edinburgh, [2]Cisco Research, [3]University College London

Task: to predict the scene classes of the **non-RGB images** based on the open texts

| | | | | | | |
|---|---|---|---|---|---|---|
| apartment | bathroom | bedroom/hotel | bookstore/library | classroom | closet | computer cluster |
| conference room | copy/mail room | dining room | game room | gym | hallway | kitchen |
| laundry room | living room/lounge | lobby | office | stairs | storage/basement/garage | misc |

## CLIP-B

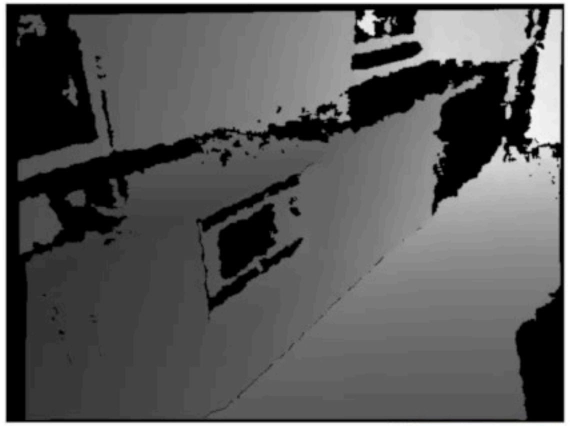


Predict

A photo of a apartment

(RGB images only for visulization)

## Ours: EdgeVL(ViT-S)

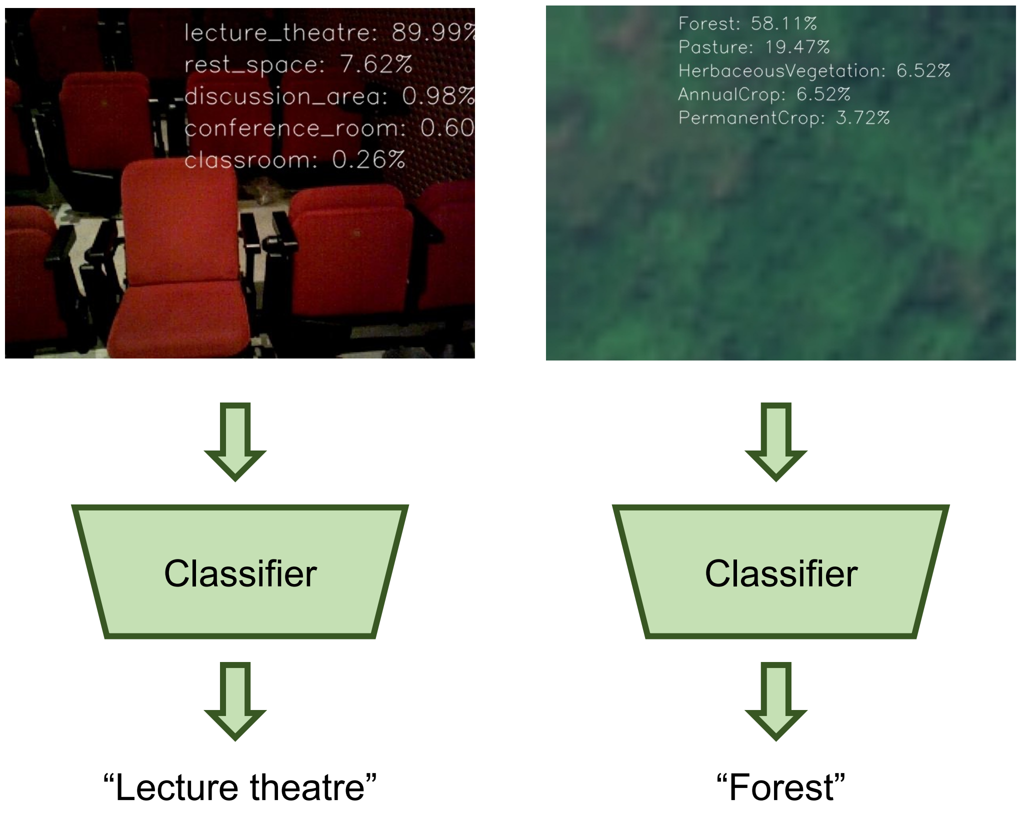


Predict

A photo of a kitchen

(RGB images only for visulization)

# Open Vocabulary Scene Classification

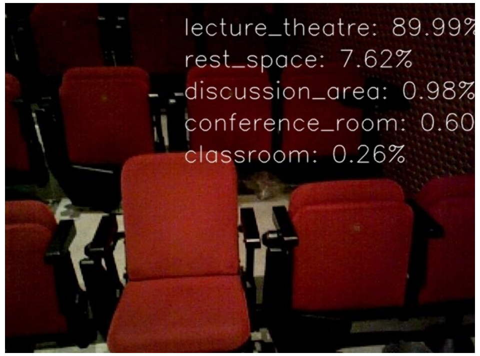



Categorizes images into a wide range of scenes, including those **not seen** during training

# Limitation 1: Modality

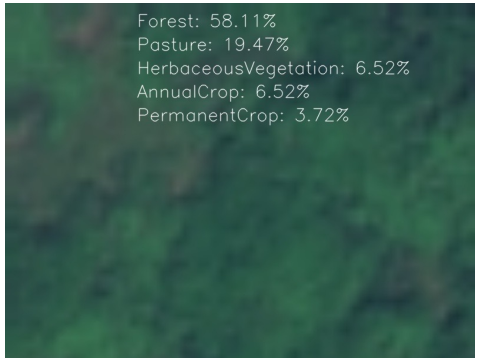
RGB modality

non-RGB modality




Lecture theatre

Correct




Forest

Correct


Rest space

Wrong


Highway

Wrong

**A Fact:** vision-language models excel in understanding RGB images but struggle with non-RGB ones.

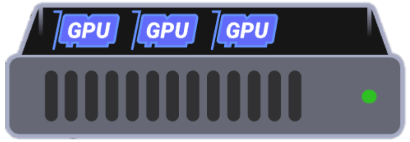
**A Question:** Can we adapt the visual embedding capabilities of vision-language models to non-RGB images while simultaneously reducing the computational footprint of the adapted model?

# Limitation 2: Computation Resource

Internet-scale images

Domain images

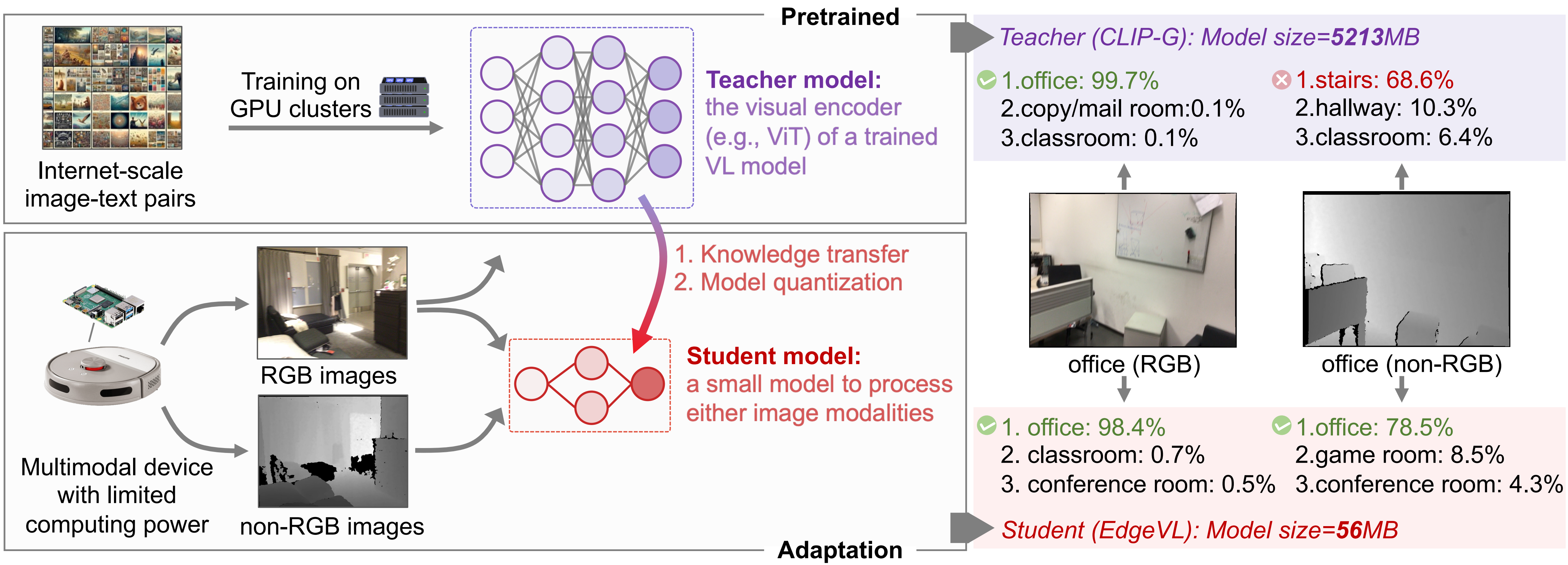
A100 cards



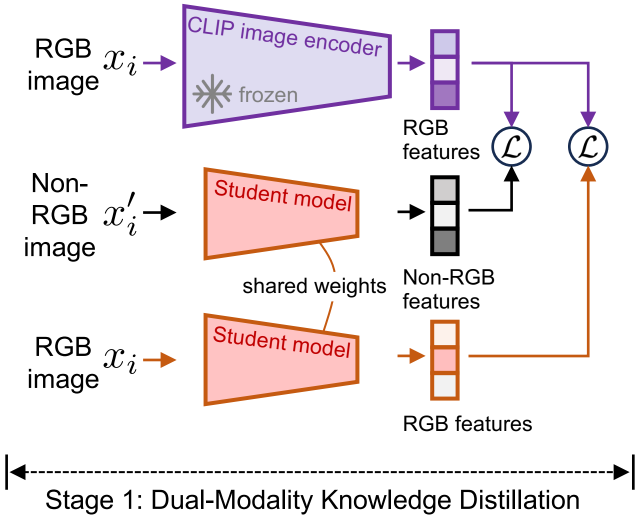

CPU

| | Vision-Language Model on a server | Model on an edge device |
|---|---|---|
| Dataset Size | Large | Small |
| Computation Resource Demand | High | Low |

# Proposed EdgeVL



# Stage-1: Dual-Modality Knowledge Distillation




## Automatic Dataset Curation

Keep samples the largest similarity scores:

$$c_i = \max\{s_k \mid s_k = \frac{e^{\Phi_{img}(x_i)^\top \Phi_{text}(y_k)}}{\sum_k^{|\mathcal{S}|} e^{\Phi_{img}(x_i)^\top \Phi_{text}(y_k)}}, k = 1, 2, \ldots, |\mathcal{S}|\},$$

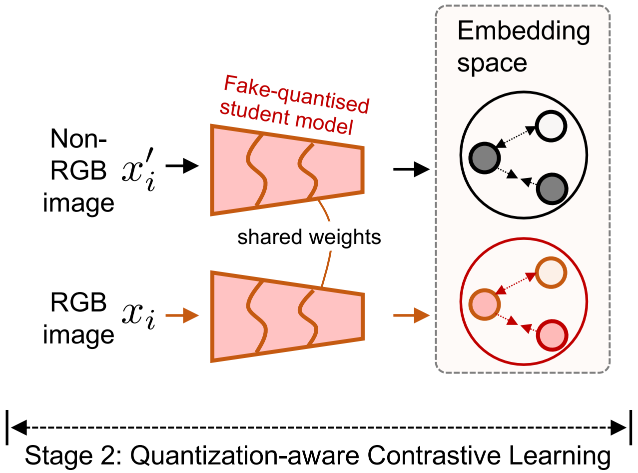
## Feature Distillation

RGB features are used as pseudo labels:

$$\mathcal{L}_d = d(\Phi_{img}(x), \Phi_{img}^{stu}(x')) + d(\Phi_{img}(x), \Phi_{img}^{stu}(x)).$$

After Stage 1: model can take either modalities as input.

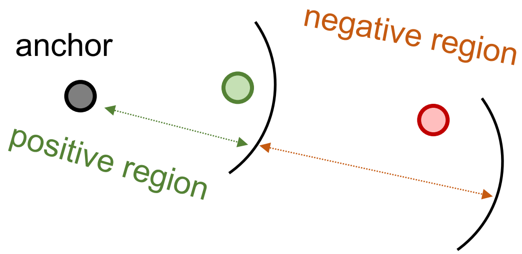
# Stage-2: Quantization-aware Contrastive Learning




## QAT Meets Contrastive Learning

Combine quantization-aware training with contrastive learning, which helps to align the embedding space.

## Triplet Sampling




semi-hard condition helps improve training convergence speed and align embeddings.

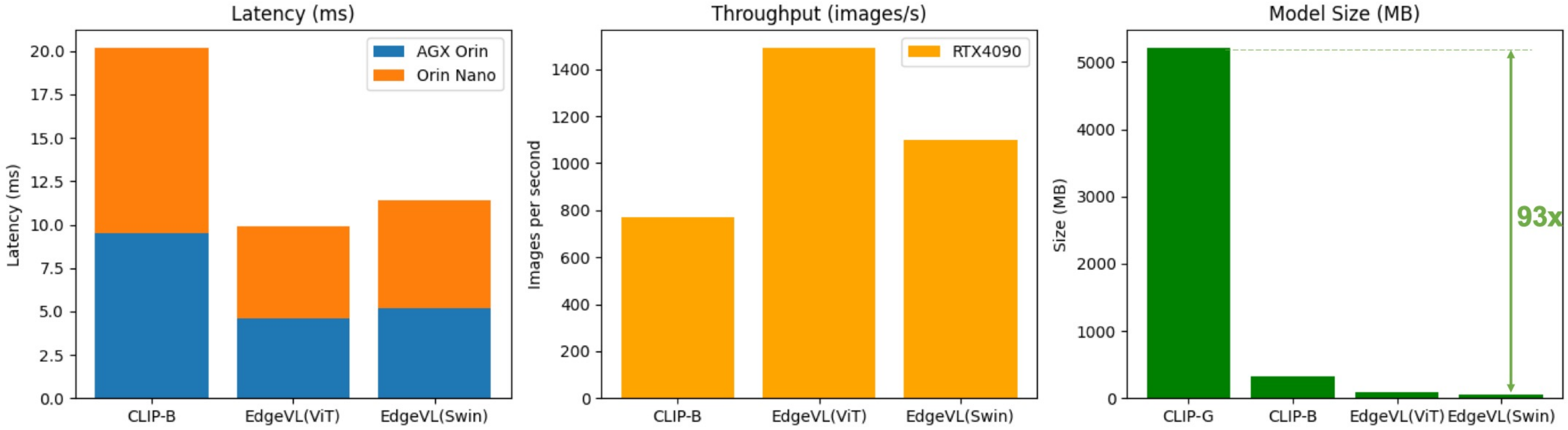
After Stage 2: accuracy increases by 15.4% while model size is 93-fold smaller.

## Performance comparison aginst SOTAs

| Methods | Bits | ScanNet (%) ↑ | | | EuroSAT (%) ↑ | | |
|---|---|---|---|---|---|---|---|
| Pretrained CLIP-B [40] | F32 | 4.5 | 36.2 | 20.4 | 16.8 | 40.4 | 28.6 |
| Pretrained CLIP-G [40] | F32 | 6.2 | 47.3 | 26.8 | 16.9 | 54.0 | 35.5 |
| Frank [17] | F32 | 8.3 | 21.7 | 15.0 | 49.2 | 37.9 | 43.5 |
| Gupta [23] | F32 | 16.0 | 17.5 | 19.8 | 54.2 | 42.4 | 48.3 |
| CMKD [24] (non-RGB) | F32 | 37.8 | 11.5 | 24.6 | 61.2 | 34.4 | 47.8 |
| CMKD [24] (RGB) | F32 | 4.0 | 42.5 | 23.2 | 20.1 | 62.4 | 41.2 |
| Fida [46] | F32 | 38.9 | 5.8 | 22.3 | 56.7 | 20.3 | 38.5 |
| CQD [45] | F32 | 40.1 | 6.7 | 23.4 | 62.4 | 36.4 | 49.4 |
| SKD [52] | F32 | 31.2 | 37.8 | 34.5 | 22.9 | 50.3 | 36.6 |
| EdgeVL (DAT-T) | Int8 | **47.9** | **52.0** | **49.9** | 61.0 | 65.7 | 63.3 |
| EdgeVL (Swin-T) | Int8 | 46.0 | 48.7 | 47.4 | 61.3 | **67.1** | 64.2 |
| EdgeVL (ViT-S) | Int8 | 42.0 | 47.5 | 44.7 | **62.9** | 66.8 | **64.8** |

EdgeVL with different backbones has higher accuracy than comparing methods.

# Efficiency comparison (using TensorRT)




EdgeVL greatly speed up the inference speed of large vision-language models.

## Ablation study: the effectiveness of Stage-1

| Methods | Bits | ScanNet (%) | | | EuroSAT (%) | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| CMKD [24] (non-RGB) | F32 | 37.8 | 11.5 | 24.6 | 61.2 | 34.4 | 47.8 |
| CMKD [24] (RGB) | F32 | 4.0 | **42.5** | 23.2 | 20.1 | **62.4** | 41.2 |
| Stage-1 (Dual-modality) | F32 | **38.6** | 40.6 | **39.6** | **61.5** | 60.3 | **60.9** |

Our dual-modality is effective in learning two modalit's features.

## Ablation study: the effectiveness of Stage-2

| Methods | Bits | DAT-T (%) | | | Swin-T (%) | | | ViT-S (%) | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | |
| Stage-1 | F32 | 38.6 | 40.6 | 39.6 | 39.9 | 41.2 | 40.5 | 37.8 | 40.7 | 39.3 |
| +PTQ [27] | Int8 | 33.0 | 36.5 | 34.8 | 29.0 | 31.7 | 30.3 | 24.7 | 25.9 | 25.3 |
| +QAT [27] | Int8 | 39.4 | 41.2 | 40.3 | 38.9 | 39.7 | 39.3 | 37.7 | 41.1 | 39.4 |
| +QViT [32] | Int8 | 35.0 | 38.0 | 36.5 | 36.5 | 38.5 | 37.5 | 31.4 | 35.3 | 33.3 |
| **+Stage-2** | Int8 | **47.9** | **52.0** | **50.0** | **46.0** | **48.7** | **47.4** | **42.0** | **47.5** | **44.7** |

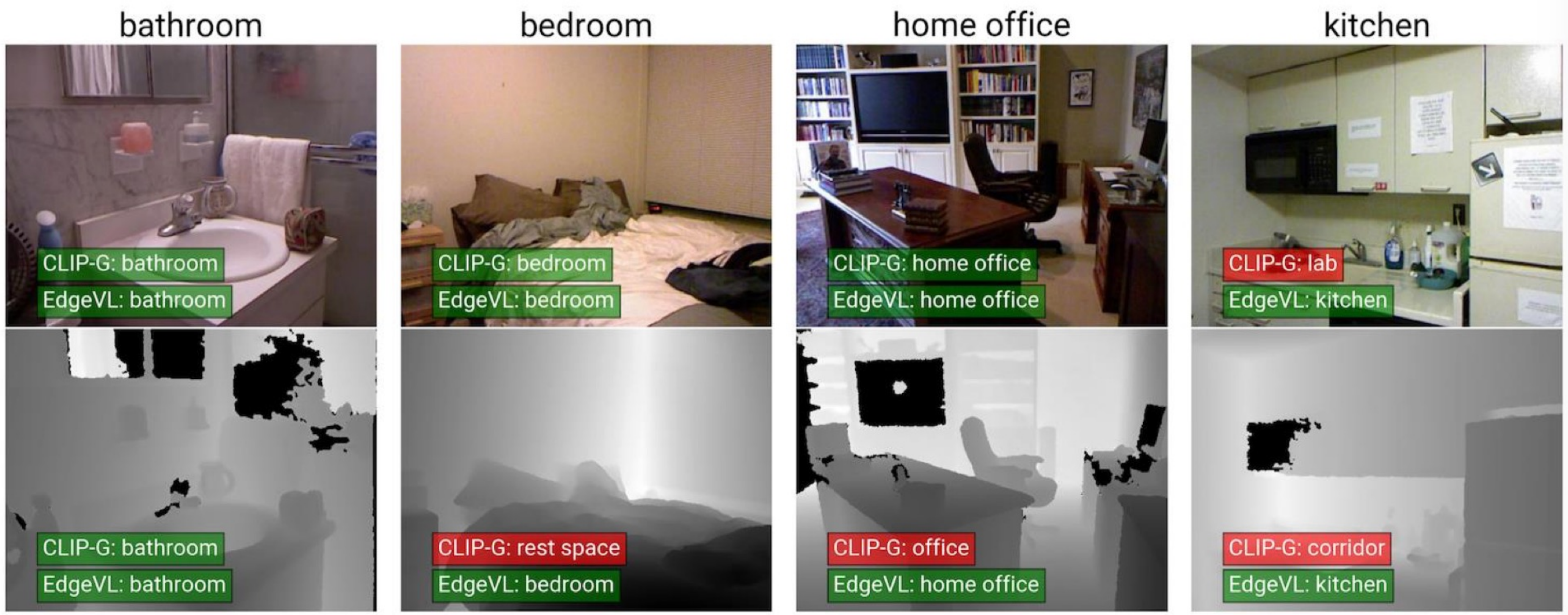
Our stage-2 is effective in improving accuracy for quantized models.

## Generelization capability

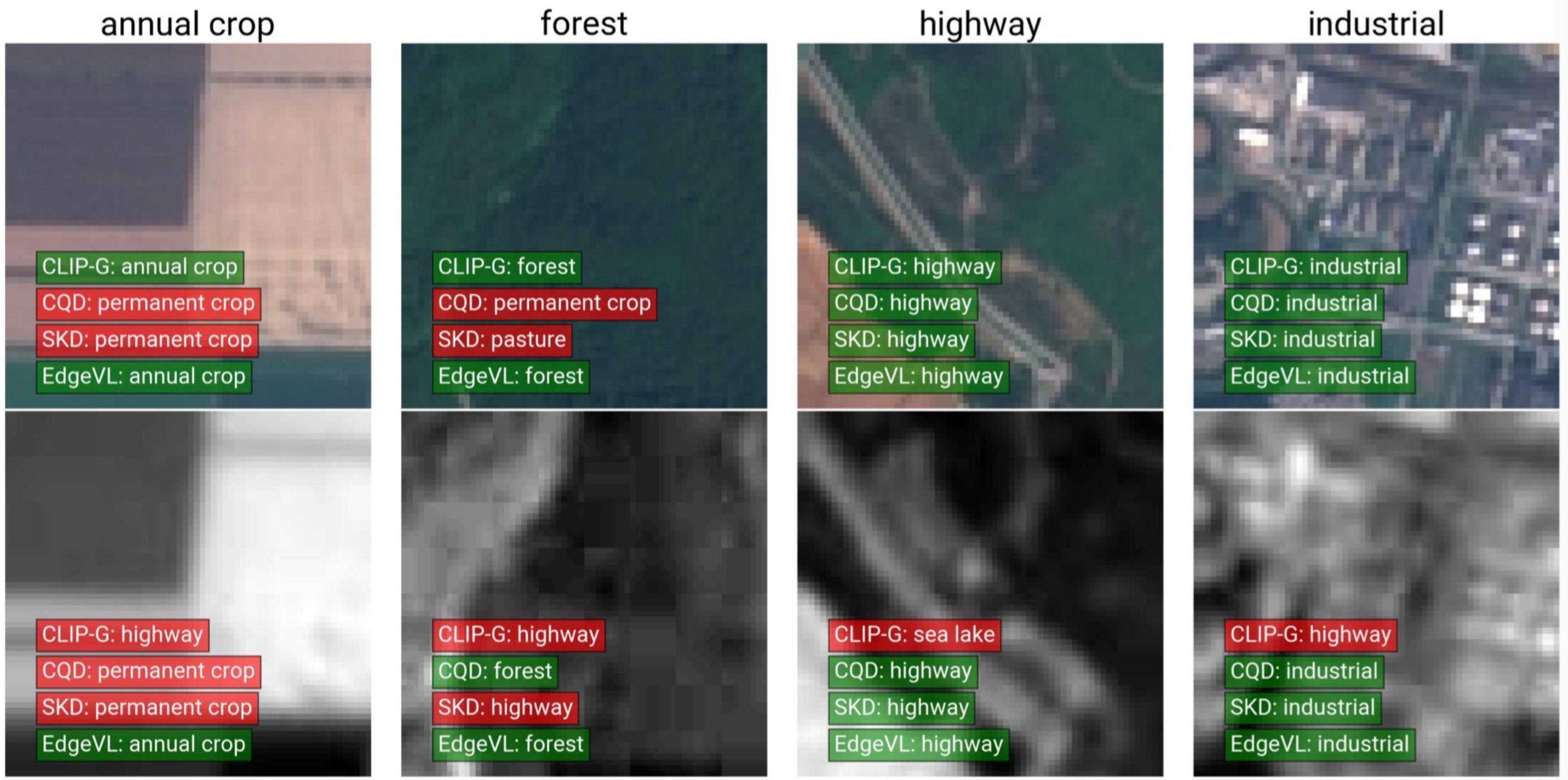
| Methods | Bits | NYU2 (%) | | | SUNRGBD (%) | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| Pre-trained CLIP-G | F32 | 25.7 | **69.7** | 47.7 | 18.0 | **54.3** | **36.2** |
| Pre-trained CLIP-B | F32 | 22.6 | 62.2 | 42.4 | 15.2 | 47.2 | 31.2 |
| **EdgeVL**: DAT-T | Int8 | **51.1** | 54.3 | **52.7** | 28.6 | 31.8 | 30.2 |
| **EdgeVL**: Swin-T | Int8 | 43.4 | 43.3 | 43.4 | **30.0** | 31.4 | 30.7 |
| **EdgeVL**: ViT-S | Int8 | 41.0 | 40.5 | 40.8 | 25.8 | 28.0 | 27.0 |

EdgeVL has comparable generalization capability than CLIP.

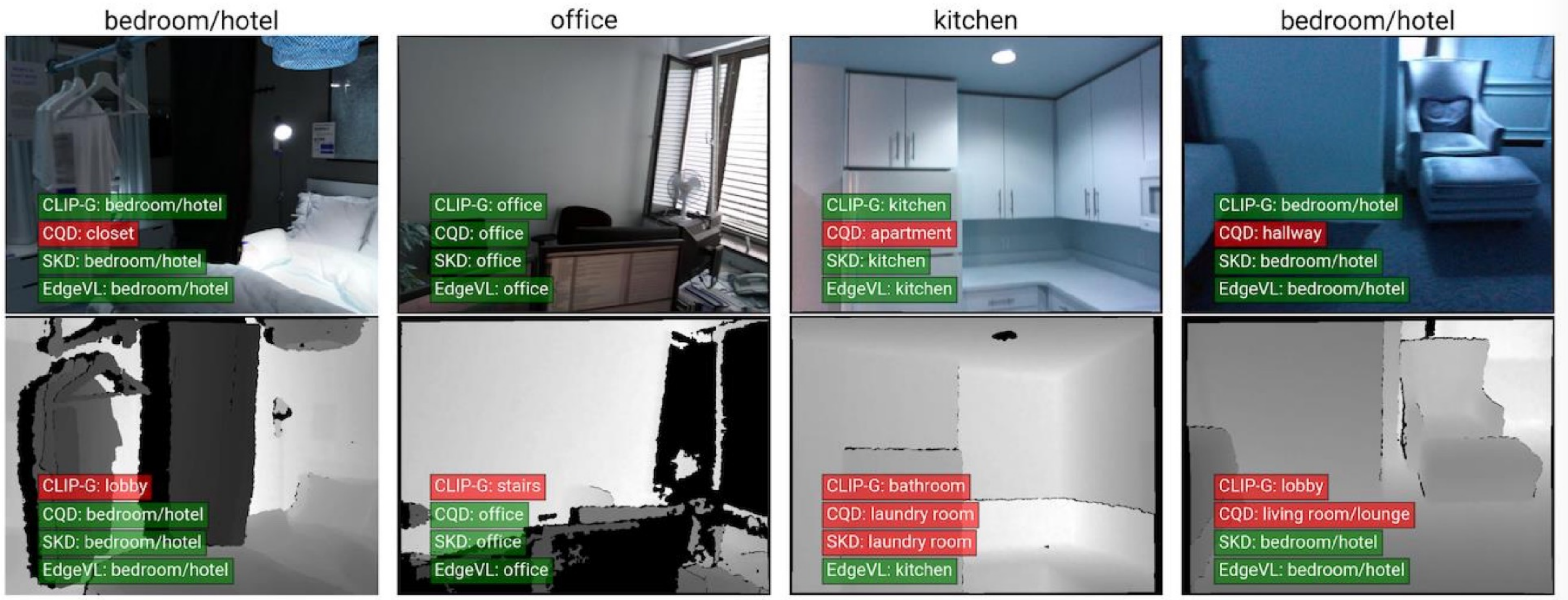
# Demonstration of EdgeVL's classification



# Demonstration of EdgeVL's classification



# Demonstration of EdgeVL's classification



## Future Works

- Enhance adaptation techniques by improving generalization performance for RGB images in crossmodal scenarios.
- Enhance the framework's versatility and effectiveness to generative vision language models.