General	ATLANTIS (Erfani et al., 2021), BDD100K (Yu et al., 2020), Dark Zurich (Sakaridis et al., 2019), DRAM (Cohen et al., 2022), FoodSeg103 (Wu et al., 2021), MHPv1 (Li et al., 2018)
Earth	FloodNet (Rahnemoonfar et al., 2020), iSAID (Zamir et al., 2019), ISPRS Potsdam (Rottensteiner et al., 2012), UAVid (Lyu et al., 2020), WorldFloods (Mateo-Garcia et al., 2021)
Medical	CHASE DB1 (Fraz et al., 2012), CryoNuSeg (Mahbod et al., 2021), Kvasir-Inst. (Jha et al., 2021), PAXRay-4 (Seibold et al., 2022)
Engineering	Corrosion CS (Bianchi & Hebdon, 2021), DeepCrack (Liu et al., 2019), PST900 (Shivakumar et al., 2019), ZeroWaste-f (Bashkirova et al., 2022)
Agriculture	CUB-200 (Wah et al., 2011), CWFID (Haug & Ostermann, 2015), SUIM (Islam et al., 2020)

Table 6: Grouping of datasets in the MESS collection (Blumenstiel et al., 2023).

A Appendix

A.1 MESS dataset composition

MESS Dataset integrates 22 datasets selected for their unique challenges, grouped into General, Earth, Medical, Engineering, and Agriculture domains. It evaluates model performance on out-of-distribution and adversarial examples, featuring visually complex medical images like those in Kvasir-Inst., and granular subclass divisions of common categories as seen in FoodSeg103 (Wu et al., 2021) and Caltech-UCSD Birds (Wah et al., 2011) datasets. Table 6 displays the dataset grouping breakdown.

A.2 Extended qualitative analysis

Figure 4 showcases additional examples where LISA encounters difficulties with certain classes in FoodSeg103. These images are selected from specific categories that proved challenging for the model. In the first image, LISA struggles to identify *mashed potato*, possibly due to its transformed state from the raw ingredient. The second image presents a biscuit-based cake, where the model incorrectly focuses on crumbs rather than recognizing the entire structure as *biscuit*. The *Hanamaki Baozi* example represents an out-of-domain concept, similar to the previously discussed Worm-eating Warbler case, highlighting the model's limitations with unfamiliar items. In the salad image, LISA misinterprets individual vegetables as the salad itself rather than recognizing the complete dish. Lastly, an adversarial example shows an apricot that visually resembles an egg, causing the model to fail in producing any output. This highlights LISA's vulnerability to visual similarities that deviate from expected appearances within a class. These examples illustrate the ongoing challenges in visual recognition tasks, particularly when dealing with transformed ingredients, culturally specific items, composite dishes, and visually ambiguous subjects.

Figure 5 presents additional visual examples of the top 10 classes that posed challenges for LISA. The *hair* class consistently proves problematic, with LISA often predicting the entire person instead of isolating the hair. For *upper clothes*, the model's misinterpretation can be attributed to linguistic ambiguity; in this instance, LISA incorrectly identified headwear as upper clothing despite being more accurately classified as an accessory. In the *soy* example, LISA fails to segment the soybean, instead erroneously detecting meatballs. The *tea* image shows the model including the cup in its segmentation rather than isolating the liquid alone. The final example demonstrates partial success, with LISA correctly identifying some cashews. However, it also exhibits a strong bias towards detecting non-relevant vegetables, leading to over-segmentation.

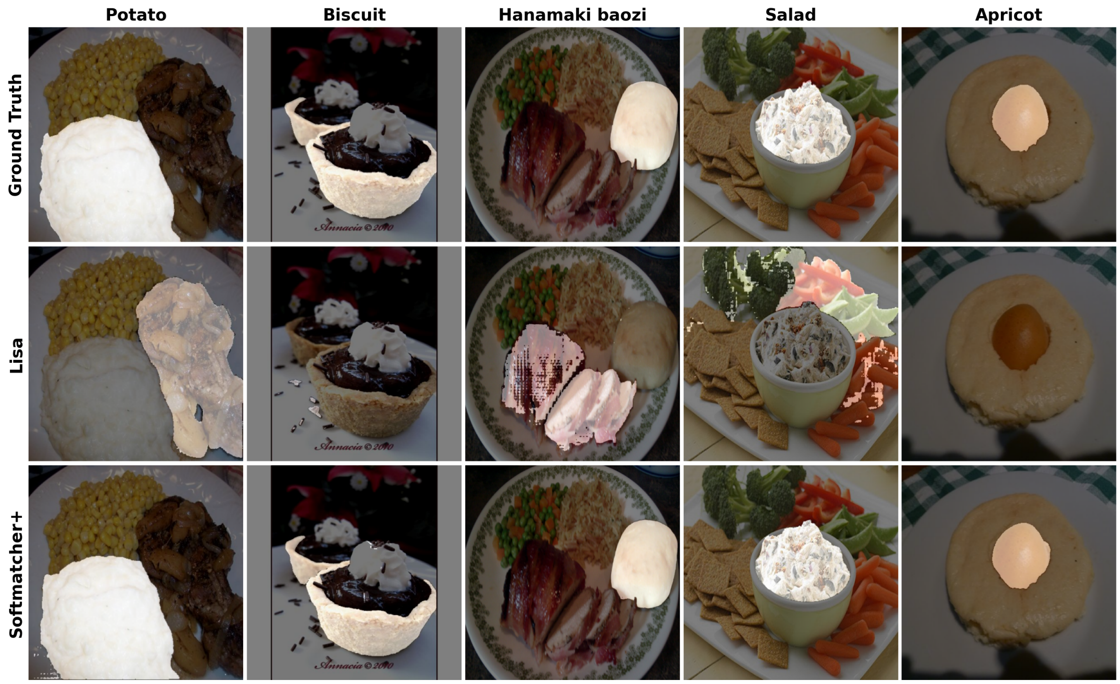

Figure 4: Qualitative examples selected from the most challenging classes of FoodSeg103.

Figure 5: Qualitative analysis on examples of challenging classes for Text Prompting.

Class name	IoU TP	IoU VP	Difference
Pole *(BDD100K)*	41.71	7.64	34.07
Fire Hydrant	33.50	0.00	33.50
Person *(ATLANTIS)*	58.33	25.77	32.56
Potted Plant	72.37	40.20	32.17
Building *(UAVid)*	66.64	34.89	31.75
White Pelican	94.32	64.48	29.84
Person *(DRAM)*	78.82	49.04	29.78
Pole *(ATLANTIS)*	33.58	4.92	28.66
Building *(Dark Zurich)*	59.75	31.49	28.26
Boat	50.98	23.50	27.48

Table 7: Top 10 classes with the highest IoU difference between text- and visual-prompted models. Results show that LISA outperforms SoftMatcher+ on classes encountered during training.

A.3 Text Prompting Superiority

We perform a mirrored analysis of Section 4.2 to better understand when LISA outperforms SoftMatcher+. Specifically, we sort the per-class IoU results and report the top 10 classes where TP surpasses VP in Table 7. Additionally, in Figure 6, we present the images with the largest difference per class for the top five classes.

Results indicate that LISA performs best in classes aligned with its training data. In fact, 9 out of 10 classes on the list appear in its training datasets (e.g., Pole, Building → ADE20K; Fire Hydrant → RefCOCOg; Person, Potted Plant, Boat → COCO). This suggests that the evaluation of these classes is largely in-domain. The alignment between test classes and training data further explains why LISA outperforms specialized models trained in-domain on "General" datasets, as pointed out in Section 3.2

On the other hand, we attribute VP's failure in these classes primarily to the broad internal variation within each category. Classes like *building* and *boat* cover a vast range of visual diversity. For instance, *boat* includes everything from freighters to rowboats, which in order to be solved a prompt optimization would be needed, in a specular way to what would be done in language. For instance, while the general term "bird" might work for identifying a *worm-eating warbler*, a more specific image prompt of a freighter would be more effective than using a general image of a *boat* for identifying a freighter.

A.4 In-Domain Performance

In this section, we explain why we intentionally avoid the traditional in-domain model performance evaluation. In Table 8, we show how our proposed method compares to LISA, SoftMatcher+, and traditional few-shot pipelines on standard few-shot semantic segmentation datasets like Pascal-5i and COCO-20i. LISA alone significantly outperforms the chosen baselines from the FSS literature and SoftMatcher+, as it was trained in-domain on the validation classes such as COCO, refCOCO and ADE20k among others. The proposed PromptMatcher, which strives to balance LISA and SoftMatcher+ doesn't reach LISA's performance levels, primarily due to the performance of the visual prompting branch, which performs significantly worse on these types of datasets than LISA.

The results support our claim that VLMs trained on massive internet-scale datasets with domains similar to (or the same as, in the case of COCO) the traditional datasets, perform exceptionally well in-domain. However, this strong in-domain performance does not translate to technical out-of-domain performance, which more closely mirrors real-world use cases. As a result, performance on traditional datasets is not a reliable indicator of the few-shot performance of the underlying model.

A.5 Extended quantitative analysis

Tables 9 and 10 present comprehensive results for text prompted and vision-only models on MESS datasets, respectively. Table 11 shows oracle results, while Table 12 displays TP-VP framework outcomes.

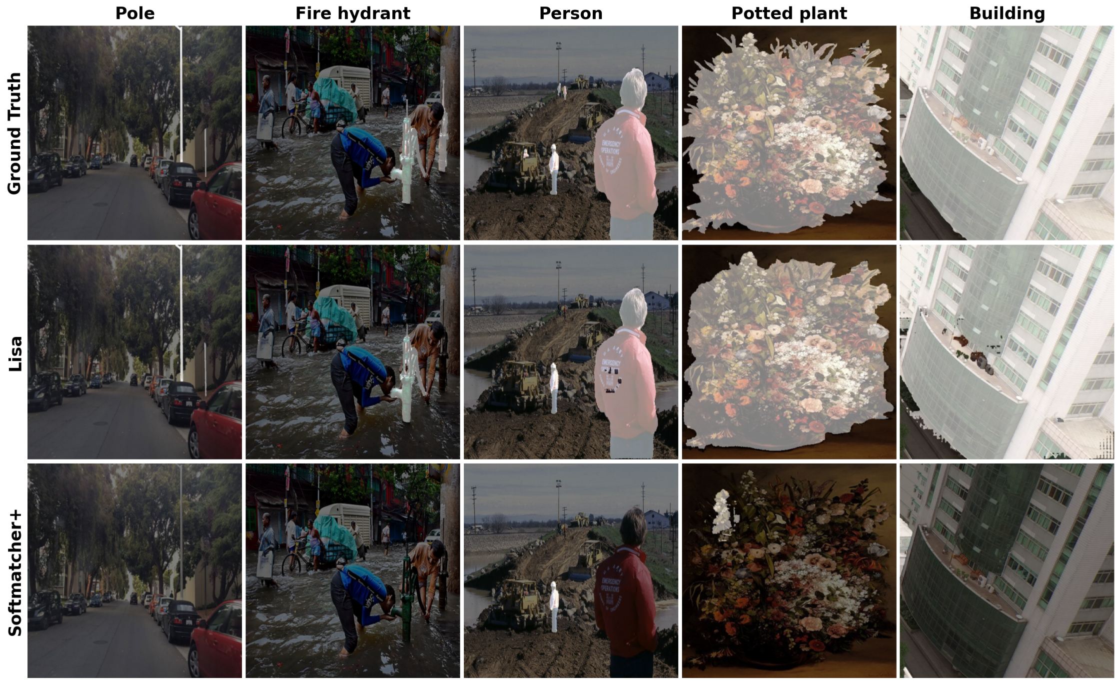

Figure 6: Qualitative analysis of examples where text prompting excels. Classes like Potted Plant and Building can vary significantly in appearance, making it challenging for SoftMatcher+ to generate accurate predictions.

Method	COCO-20^i	Pascal-5^i
Painter	32.80	64.50
Seggpt	56.10	**83.20**
PAGMA-Net *(CLIP-RN101)*	**59.40**	77.60
HMNet	52.10	70.40
LISA	**72.23**	**80.97**
SoftMatcher+	55.12	67.98
PromptMatcher	59.07	77.13

Table 8: In-domain performance on FSS Datasets.

A.6 PromptMatcher Pseudocode

Algoritm 1 showcases PromptMatcher pseudocode.

Algorithm 1: PromptMatcher

Input: *reference_image*, *reference_mask*, *reference_text*,*target_image*
Output: *final_mask*

ref_feats ← extract_features(*reference_image*); `// Extract Features`
targ_feats ← extract_features(*target_image*);
targ_sam_feats ← extract_SAM_features(*target_image*);

probability_map ← match_features(*ref_feats*, *reference_mask*, *targ_feats*); `// SoftMatcher+`
prompt_points ← sample_and_cluster(*probability_map*);
softmatcher_masks ← SAM_mask_decoder(*prompt_points*, *target_sam_feats*);

lisa_SEG_token ← LISA_VLM(*target_image*, *reference_text*); `// LISA`
lisa_mask ← LISA_mask_decoder(*target_sam_feats*, *LISA_SEG_token*) ;

mask_proposals ← *lisa_mask* + *mask_proposals* ; `// Merge masks`

masks ← reject_masks(*mask_proposals*) ; `// Reject and merge masks`
segmentation_mask ← merge_masks(*masks*);

return *segmentation_mask*

	Dataset	SEEM txt	CAT-Seg	Florence	PALI-Gem	NACLIP	LISA	Supervised
General	ATLANTIS	48.4	30.5	14.4	46.8	46.79	63.9	45.1
	BDD100K	32.6	30.6	4.5	25.9	27.54	78.0	82.3
	Dark Zurich	33.1	45.8	11.4	21.8	34.37	41.1	44.8
	DRAM	60.4	33.6	29.3	58.6	50.05	78.6	42.2
	FoodSeg103	31.0	30.0	18.1	51.3	37.81	60.6	53.2
	MHP v1	10.0	33.1	6.5	7.6	19.77	19.8	63.9
Earth	FloodNet	59.6	9.2	28.6	62.5	66.35	72.9	84.6
	iSAID	9.5	66.5	4.1	4.3	9.80	31.3	45.7
	ISPRS Potsdam	40.7	53.9	11.0	23.9	39.36	41.0	74.0
	UAVid	57.5	39.0	11.5	34.7	56.44	59.8	87.2
	WorldFloods	16.9	16.1	14.4	20.3	33.94	33.4	65.3
Medical	CHASE DB1	9.8	49.9	9.1	8.9	10.05	16.7	92.7
	CryoNuSeg	24.1	39.8	6.7	24.2	24.77	31.9	82.2
	Kvasir-Inst.	28.6	51.4	10.2	44.9	12.97	23.2	87.6
	PAXRay-4	53.1	42.0	26.7	35.7	43.11	54.9	67.8
Engin.	Corrosion CS	11.1	25.0	7.7	8.8	4.47	13.8	97.1
	DeepCrack	4.2	35.1	5.5	4.5	4.78	6.8	73.5
	PST900	14.3	79.4	6.3	2.9	3.87	12.1	93.7
	ZeroWaste-f	26.2	54.5	9.8	12.9	13.93	18.5	93.8
Agri.	CUB-200	89.0	31.4	0.0	68.2	14.36	88.1	85.9
	CWFID	13.7	25.3	4.2	7.0	11.79	36.6	52.5
	SUIM	31.0	16.9	18.7	44.9	40.86	67.2	49.9

Table 9: Per dataset performance of text prompted methods

	Dataset	SEEM vis	DINOv	VP	SoftMatcher+	Supervised
General	ATLANTIS	15.8	52.8	45.0	51.4	45.1
	BDD100K	7.2	37.8	53.1	58.5	82.3
	Dark Zurich	4.0	22.6	45.4	47.7	44.8
	DRAM	13.4	73.6	55.9	62.9	42.2
	FoodSeg103	11.8	28.3	54.0	60.5	53.2
	MHP v1	5.6	9.5	34.6	36.7	63.9
Earth	FloodNet	41.6	59.9	56.7	57.4	84.6
	iSAID	2.2	4.3	22.8	26.7	45.7
	ISPRS Potsdam	13.0	24.2	41.2	41.4	74.0
	UAVid	15.5	34.5	32.7	35.7	87.2
	WorldFloods	11.9	17.3	16.4	20.0	65.3
Medical	CHASE DB1	10.4	9.6	0.0	0.0	92.7
	CryoNuSeg	26.8	24.0	21.2	24.5	82.2
	Kvasir-Inst.	6.5	24.4	65.7	58.0	87.6
	PAXRay-4	38.1	39.0	39.0	39.1	67.8
Engin.	Corrosion CS	9.3	10.1	7.2	14.8	97.1
	DeepCrack	3.6	4.5	30.7	39.3	73.5
	PST900	4.5	4.8	16.4	38.9	93.7
	ZeroWaste-f	10.4	13.9	21.0	21.9	93.8
Agri.	CUB-200	20.7	92.2	85.4	87.0	85.9
	CWFID	17.5	33.5	41.5	41.0	52.5
	SUIM	26.9	51.4	52.5	54.1	49.9

Table 10: Per dataset performance of visual prompted methods

	Dataset	SoftMatcher+	LISA	Oracle	Oracle+	Supervised
General	ATLANTIS	51.4	63.9	63.9	68.9	45.1
	BDD100K	58.5	78.0	78.0	79.2	82.3
	Dark Zurich	47.7	41.1	47.7	55.0	44.8
	DRAM	62.9	78.6	78.6	81.3	42.2
	FoodSeg103	60.5	60.6	60.6	74.0	53.2
	MHP v1	36.7	19.8	36.7	45.3	63.9
Earth	FloodNet	57.4	72.9	72.9	74.8	84.6
	iSAID	26.7	31.3	31.3	35.4	45.7
	ISPRS Potsdam	41.4	41.0	41.4	50.2	74.0
	UAVid	35.7	59.8	59.8	65.0	87.2
	WorldFloods	20.0	33.4	33.4	33.4	65.3
Medical	CHASE DB1	0.0	16.7	16.7	16.7	92.7
	CryoNuSeg	24.5	31.9	31.9	34.5	82.2
	Kvasir-Inst.	58.0	23.2	58.0	72.0	87.6
	PAXRay-4	39.1	54.9	54.9	61.7	67.8
Engin.	Corrosion CS	14.8	13.8	14.8	17.6	97.1
	DeepCrack	39.3	6.8	39.3	42.2	73.5
	PST900	38.9	12.1	38.7	39.7	93.7
	ZeroWaste-f	21.9	18.5	21.9	30.5	93.8
Agri.	CUB-200	87.0	88.1	88.1	90.5	85.9
	CWFID	41.0	36.6	41.0	48.4	52.5
	SUIM	54.1	67.2	67.2	75.2	49.9

Table 11: Per dataset performance of Oracle ensembling baselines.

	Dataset	SEEM	LISA	SoftMatcher+	PromptMatcher	Oracle+	Supervised
General	ATLANTIS	15.8	63.9	51.4	55.7	68.9	45.1
	BDD100K	6.9	78.0	58.5	67.3	79.2	82.3
	Dark Zurich	4.3	41.1	47.7	51.7	55.0	44.8
	DRAM	13.5	78.6	62.9	69.7	81.3	42.2
	FoodSeg103	12.0	60.6	60.7	61.9	74.0	53.2
	MHP v1	5.8	19.8	36.7	46.2	45.3	63.9
Earth	FloodNet	40.7	72.9	57.4	61.4	74.8	84.6
	iSAID	2.3	31.3	26.7	24.3	35.4	45.7
	ISPRS Potsdam	13.1	41.0	41.4	45.9	50.2	74.0
	UAVid	14.9	59.8	35.7	52.4	65.0	87.2
	WorldFloods	14.2	33.4	20.0	14.7	33.4	65.3
Medical	CHASE DB1	10.4	16.7	0.0	0.0	16.7	92.7
	CryoNuSeg	27.1	31.9	24.5	24.1	34.5	82.2
	Kvasir-Inst.	6.4	23.2	58.0	60.8	72.0	87.6
	PAXRay-4	38.1	54.9	39.1	55.5	61.7	67.8
Engin.	Corrosion CS	10.4	13.8	14.8	15.2	17.6	97.1
	DeepCrack	3.8	6.8	39.3	42.6	42.2	73.5
	PST900	4.9	12.1	38.9	39.3	39.9	93.7
	ZeroWaste-f	10.1	18.5	21.9	24.6	30.5	93.8
Agri.	CUB-200	21.1	88.1	87.0	88.9	90.5	85.9
	CWFID	17.5	36.6	41.0	38.4	48.4	52.5
	SUIM	28.8	67.2	54.1	59.8	75.2	49.9

Table 12: Per dataset performance of visual-text prompted methods