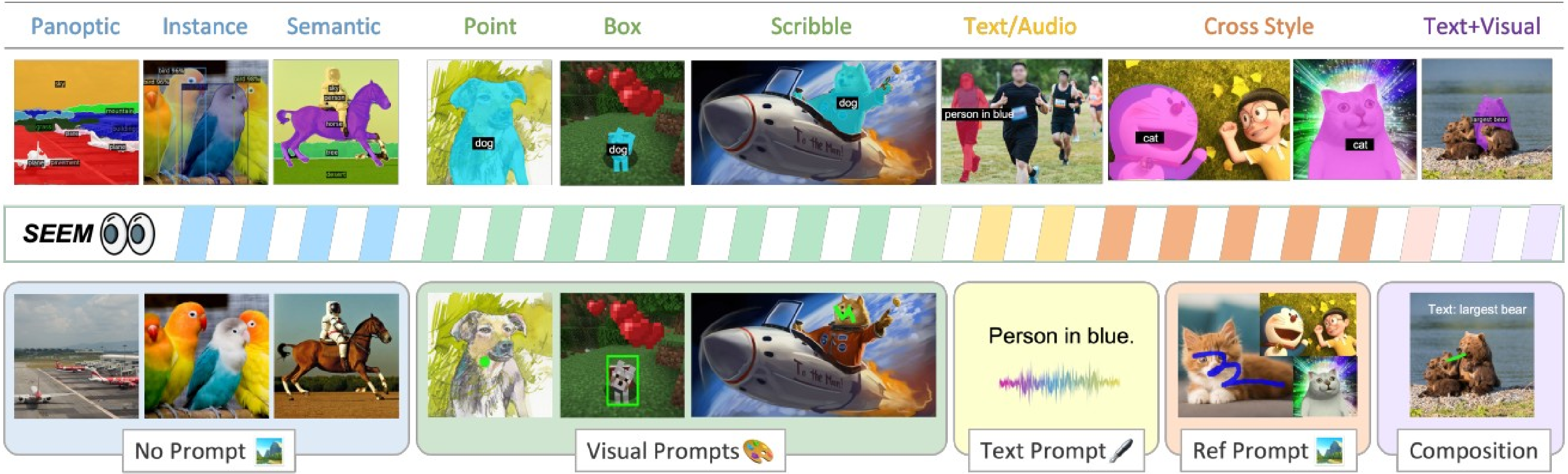
SEEM: Segment Everything Everywhere All at Once

Han Gao

Nov. 21st, 2023

Introduction

SEEM is a novel decoding mechanism that enables diverse prompting for all types of segmentation tasks, aiming at a universal segmentation interface that behaves like large language models (LLMs).


Panoptic
Instance
Semantic
Point
Box
Scribble
Text/Audio
Cross Style
Text+Visual
dog
dog
dog
person in blue
cat
cat
SEEM
Person in blue.
Text: largest bear
No Prompt
Visual Prompts
Text Prompt
Ref Prompt
Composition

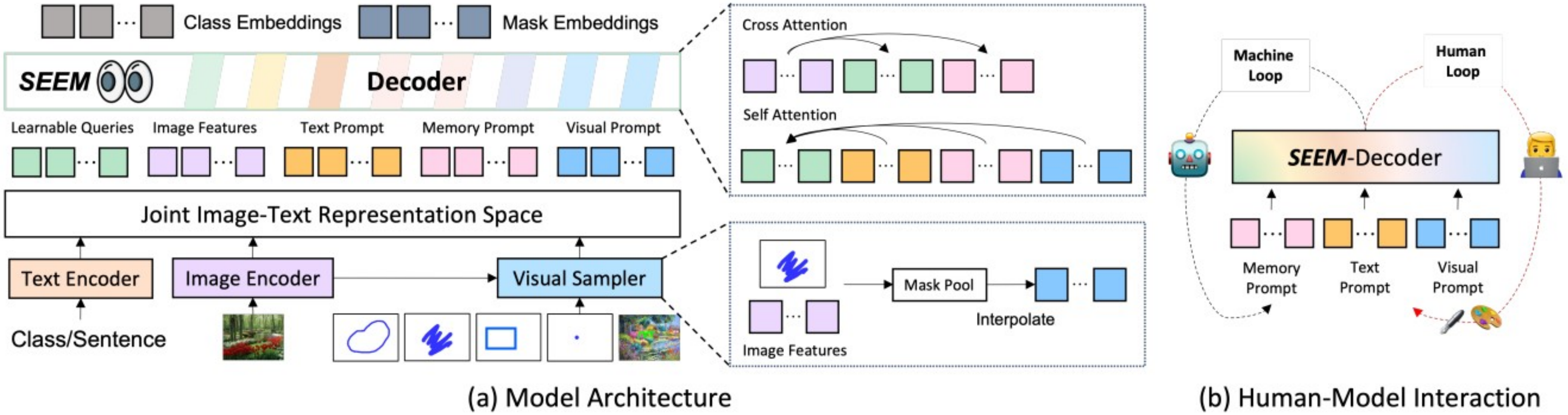
Properties

Versatility
Compositionality
Interactivity
Semantic-awareness

Model design

SEEM employs a generic encoder-decoder architecture but also employs a sophisticated interaction scheme between queries and prompts.

Architecture



Mathematical expression

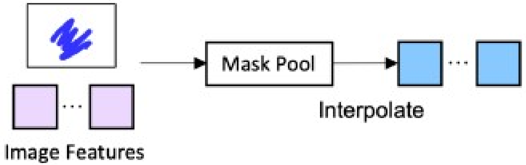
$$\langle O_h^m, O_h^c \rangle = Decoder(Q_h; \langle P_t, P_v, P_m \rangle | Z)$$

$$M = MaskPredictor(O_h^m)$$

$$C = ConceptClassifier(O_c^m)$$

Versatility

There are visual prompts to handle all non-textual inputs. These non-textual queries are beneficial to disambiguate the user's intent when textual prompts alone fail to identify the correct segment.



$$P_v = VisualSampler(s, \hat{Z})$$

$$s \in \{points, box, scribbles, polygons\}$$

Compositionality

For real-world applications, a compositional approach to prompting is essential. However, the training data usually only covers a single type of interaction, and the embedding spaces remain inherently different. To solve the problems, matching different types of prompts with different outputs is useful. In particular, simple concatenate is fine.

Interactivity

Interactive segmentation usually cannot be completed in one shot and requires multiple interaction rounds for refinement, therefore memory prompts, which encode the history information by using a mask-guided cross-attention layer, are used

$$P_m^l = MaskedCrossAtt(P_m^{l-1}; M_p|Z)$$

Semantic-awareness

SEEM produces semantic labels to masks for all kinds of prompt combinations in a zero-shot manner, since the visual prompt features are aligned with textual features in a joint visual-semantic space.

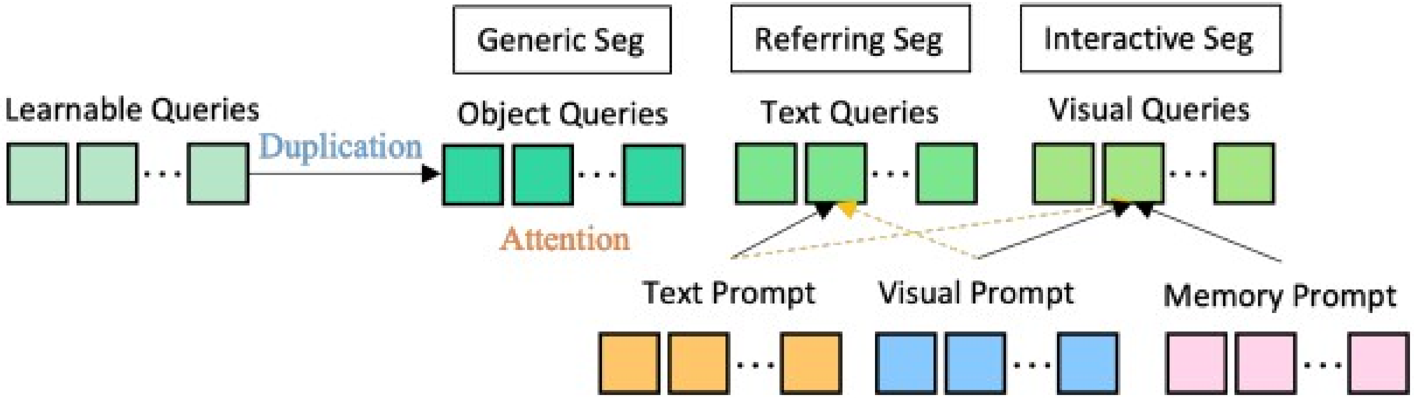
Pseudo code

Algorithm 1: Pseudo code for SEEM.

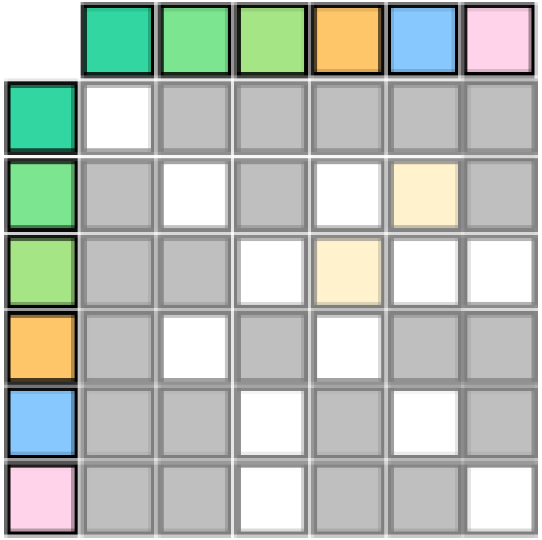
```
# Inputs: Image(img)[B,3,H,W]; Pos_Mask(pm), Neg_Mask(nm)[B,1,H,W]; Text(txt)[abc...];
# Variables: Learnable Queries(Q_h); Attention Masks between Q and P(qpm)
# Functions: Img_Encoder(),Text_Encoder(),Visual_Sampler(),feature_attn(),prompt_attn(),output();

def init( ):
    Q_o,Q_t,Q_v = Q_h.copy();# Initialize object, text and visual queries.
    F_v,P_t = Img_Encoder(img), Text_Encoder(txt);# F_v and P_t denote image feature, text
      prompt.
    P_v = Visual_Sampler(F_v, pm, nm);# Sample visual prompt from image feature, pos/neg
      mask.
def SEEM_Decoder(F_v,Q_o,Q_t,Q_v,P_v,P_t,P_m):
    Q_o,Q_t,Q_v = feature_attn(F_v,Q_o,Q_t,Q_v);# Cross attend queries with image features.
    Q_o,Q_t,Q_v = prompt_attn(qpm,Q_o,Q_t,Q_v,P_v,P_t,P_m);# Self attend queries and prompts.
    O_m,O_c,P_m = output(F_v,Q_o,Q_t,Q_v);# Compute mask and class outputs.
def forward(img,pm,nm,txt):
    F_v,Q_o,Q_t,Q_v,P_v,P_t = init(); P_m = None;# Initialize variables.
    for i in range(max_iter):
        O_m,O_c,P_m = SEEM_Decoder(F_v,Q_o,Q_t,Q_v,P_v,P_t,P_m)
```

Cross attend queries with image features



Self-attend queries and prompts



Loss function

SEEM is trained with a linear combination of losses for panoptic segmentation, referring segmentation, and interactive segmentation.

$$L = \begin{bmatrix} L_{c\_CE\_pano} & L_{m\_BCE\_pano} & L_{m\_DICE\_pano} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} + \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} L_{c\_CE\_ref} & L_{m\_BCE\_ref} & L_{m\_DICE\_ref} \\ L_{c\_CE\_iseg} & L_{m\_BCE\_iseg} & L_{m\_DICE\_iseg} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

Experiments

SEEM is trained on three tasks: panoptic segmentation, referring segmentation, and interactive segmentation. The framework follows X-Decoder except decoder, which includes a vision backbone, a language backbone, an encoder, and the SEEM-decoder.

Training datasets:

Panoptic segmentation: COCO2017 with panoptic segmentation annotations

Referring segmentation: combination of Ref-COCO, Ref-COCOg, and Ref-COCO+ for COCO image annotations

Interactive segmentation: COCO2017 with panoptic segmentation annotations

Evaluation metrics

Panoptic segmentation: PQ (Panoptic Quality)
Instance segmentation: AP (Average Precision)
Semantic segmentation: mIoU (mean Intersection over Union)
Interactive segmentation: Number of Clicks (NoC)

Results

Method	Segmentation Data	Type	Generic Segmentation COCO			Referring Segmentation RefCOCOg			Interactive Segmentation PascalVOC					
			PQ	mAP	mIoU	cIoU	mIoU	AP50	5-NoC85	10-NoC85	20-NoC85	5-NoC90	10-NoC90	20-NoC90
Mask2Former (T) [6]	COCO (0.12M)		53.2	43.3	63.2	-	-	-	-	-	-	-	-	-
Mask2Former (B) [6]	COCO (0.12M)		56.4	46.3	67.1	-	-	-	-	-	-	-	-	-
Mask2Former (L) [6]	COCO (0.12M)		57.8	48.6	67.4	-	-	-	-	-	-	-	-	-
Pano/SegFormer (B) [45]	COCO (0.12M)	Segmentation	55.4	*	*	-	-	-	-	-	-	-	-	-
LAVT (B) [53]	Ref-COCO (0.03M)		-	-	-	61.2	*	*	-	-	-	-	-	-
PolyFormer (B) [17]	Ref-COCO+VG+... (0.16M)		-	-	-	69.3	*	*	-	-	-	-	-	-
PolyFormer (L) [17]	Ref-COCO+VG+... (0.16M)		-	-	-	71.1	*	*	-	-	-	-	-	-
RITM (<T) [18]	COCO+LVIS (0.12M)		-	-	-	-	-	-	*	*	2.19	*	*	2.57
PseudoClick (<T) [54]	COCO (0.12M)		-	-	-	-	-	-	*	*	1.94	*	*	2.25
FocalClick (T) [21]	COCO (0.12M)		-	-	-	-	-	-	*	*	2.97	*	*	3.52
FocalClick (B) [21]	COCO (0.12M)	Interactive	-	-	-	-	-	-	*	*	2.46	*	*	2.88
SimpleClick (B) [20]	COCO+LVIS (0.12M)		-	-	-	-	-	-	1.75	1.93	2.06	1.94	2.19	2.38
SimpleClick (L) [20]	COCO+LVIS (0.12M)		-	-	-	-	-	-	1.52	1.64	1.72	1.67	1.84	1.96
SimpleClick (H) [20]	COCO+LVIS (0.12M)		-	-	-	-	-	-	1.51	1.64	1.76	1.64	1.83	1.98
UViM (L) [55]	COCO (0.12M)		45.8	*	*	-	-	-	-	-	-	-	-	-
Pix2Seq v2 (B) [56]	COCO (0.12M)		-	38.2	-	-	-	-	-	-	-	-	-	-
X-Decoder (T) [11]	COCO (0.12M)		52.6	41.3	62.4	59.8	*	*	-	-	-	-	-	-
X-Decoder (B) [11]	COCO (0.12M)		56.2	45.8	66.0	64.5	*	*	-	-	-	-	-	-
X-Decoder (L) [11]	COCO (0.12M)		56.9	46.7	67.5	64.6	*	*	-	-	-	-	-	-
UNINEXT (T) [48]	Image+Video (3M)		-	44.9	-	70.0	*	*	-	-	-	-	-	-
UNINEXT (L) [48]	Image+Video (3M)		-	49.6	-	73.4	*	*	-	-	-	-	-	-
Painter (L) [57]	COCO+ADE+NYUv2 (0.16M)	Generalist	43.4	*	*	-	-	-	-	-	-	-	-	-
#SegGPT (L) [50]	COCO+ADE+NYUv2 (0.16M)		34.4	*	*	-	-	-	-	-	-	-	-	-
#SAM (B) [36]	SAM (11M)		-	-	-	-	-	-	2.47	2.65	3.28	2.23	3.13	4.12
#SAM (L) [36]	SAM (11M)		-	-	-	-	-	-	1.85	2.15	2.60	2.01	2.46	3.12
#SAM (H) [36]	SAM (11M)		-	-	-	-	-	-	1.82	2.13	2.55	1.98	2.43	3.11
SEEM (T)	COCO+LVIS (0.12M)		50.8	39.7	62.2	60.9	65.7	74.8	1.72	2.30	3.37	1.97	2.83	4.41
SEEM (B)	COCO+LVIS (0.12M)		56.1	46.4	66.3	65.0	69.6	78.2	1.56	2.04	2.93	1.77	2.47	3.79
SEEM (L)	COCO+LVIS (0.12M)		57.5	47.7	67.6	65.6	70.3	78.9	1.51	1.95	2.77	1.71	2.36	3.61
SEEM (T)	COCO+LVIS (0.12M)		-	-	-	70.4	71.7	82.1	1.72	2.28	3.32	1.97	2.82	4.37
SEEM (B)	COCO+LVIS (0.12M)	Composition	-	-	-	76.2	77.8	87.8	1.56	2.03	2.91	1.77	2.46	3.76
SEEM (L)	COCO+LVIS (0.12M)		-	-	-	75.1	76.9	86.8	1.52	1.97	2.81	1.72	2.38	3.64

Results

Generic segmentation: SEEM maintains competitive panoptic, instance, and semantic segmentation performance.

Referring segmentation: SEEM achieves competitive performance. By adding a visual compositional prompt, performance is better.

Results

Interactive segmentation: SEEM provides strong compositional capabilities and is more efficient.

Method	COCO					Open Image					ADE				
	Point 1-IoU	Stroke 1-IoU	Scribble 1-IoU	Polygon 1-IoU	Box 1-IoU	Point 1-IoU	Stroke 1-IoU	Scribble 1-IoU	Polygon 1-IoU	BoX 1-IoU	Point 1-IoU	Stroke 1-IoU	Scribble 1-IoU	Polygon 1-IoU	BoX 1-IoU
SimpleClick (B)	49.0	33.1	65.1	48.6	42.5	48.6	29.5	54.2	49.5	42.7	47.0	19.0	52.1	48.3	37.2
SimpleClick (L)	38.9	33.9	68.8	39.2	34.7	37.5	29.1	59.8	35.2	31.2	36.8	16.4	56.4	41.7	29.5
SimpleClick (H)	59.0	37.3	71.5	45.3	52.4	54.1	32.6	64.7	39.9	49.3	52.8	18.4	58.3	46.8	41.8
SAM (B)	58.6	22.8	34.2	44.5	50.7	62.3	28.4	39.2	45.8	53.6	51.0	21.9	31.1	31.0	**58.8**
SAM (L)	64.7	44.4	57.1	60.7	50.9	65.3	45.9	55.7	57.8	52.4	57.4	45.8	53.1	45.8	58.7
SAM (H)	65.0	27.7	30.6	37.8	50.4	67.7	26.5	29.9	41.9	52.1	58.4	20.4	22.2	28.3	58.5
SEEM (T)	78.9	81.0	81.2	72.2	73.7	67.1	**69.4**	**69.5**	63.1	**60.9**	65.4	67.3	67.3	59.0	53.4
SEEM (B)	81.7	82.8	83.5	76.0	75.7	**67.6**	69.0	68.7	**64.2**	60.3	**66.4**	**68.6**	**67.7**	**60.5**	53.6
SEEM (L)	**83.4**	**84.6**	**84.1**	**76.5**	**76.9**	66.8	67.8	67.6	62.4	60.1	65.5	66.6	66.3	58.1	54.1

Results

Video object segmentation(zero-shot): SEEM can do video object segmentation in a zero-shot manner without training with video or pairwise image data. The future is pro

Method	Segmentation Data	Type	Refer-Type	Zero-Shot	Single Image	DAVIS17 JF	DAVIS17 J	DAVIS17 F	DAVIS16-Interactive JF	DAVIS16-Interactive J	DAVIS16-Interactive F	YouTube-VOS 2018 G	YouTube-VOS 2018 Js	YouTube-VOS 2018 Fs	YouTube-VOS 2018 Ju	YouTube-VOS 2018 Fu
With Video Data																
AGSS [63]	VOS+DAVIS (0.1M)	Video	Mask	✗	✗	67.4	64.9	69.9	-	-	-	71.3	71.3	65.5	75.2	73.1
AGAME [64]	(Synth)VOS+DAVIS (0.11M)	Video	Mask	✗	✗	70.0	67.2	72.7	-	-	-	66.0	66.9	*	61.2	*
SWEM [65]	Image+VOS+DAVIS (0.25M)	Video	Mask	✗	✗	84.3	81.2	87.4	-	-	-	82.8	82.4	86.9	77.1	85.0
XMem [66]	Image+VOS+DAVIS (0.25M)	Video	Mask	✗	✗				-	-	-	86.1	85.1	89.8	80.3	89.2
SiamMask [67]	COCO+VOS (0.21M)	Video	Box	✗	✗	*	54.3	58.5	69.8	71.7	67.8	*	60.2	58.2	45.1	47.7
MiVOS [19]	BL30K+VOS+DAVIS (4.88M)	Video	Mask/Scribble	✗	✗	84.5	81.7	87.4	91.0	89.6	92.4	82.6	81.1	85.6	77.7	86.2
ReferFormer-B [68]	RefCOCO(+/g)+VOS+DAVIS (0.13M)	Video	Text	✗	✗	61.1	58.1	64.1	-	-	-	*	*	*	*	*
TAM-L [69]	XMem+SAM (11.2M)	Generalist	Multiple Points	✗	✗	-	-	-	88.4	87.5	89.4	-	-	-	-	-
UNINEXT-T [48]	Image+Video (3M)	Generalist	Mask	✗	✗	74.5	71.3	77.6	-	-	-	77.0	76.8	81.0	70.8	79.4
UNINEXT-L [48]	Image+Video (3M)	Generalist	Mask	✗	✗	77.2	73.2	81.2	-	-	-	78.1	79.1	83.5	71.0	78.9
UNINEXT-L [48]	Image+Video (3M)	Generalist	Text	✗	✗	66.7	62.3	71.1	-	-	-	*	*	*	*	*
Without Video Data																
Painter-L [57]	COCO+ADE+NYUv2 (0.16M)	Generalist	Mask	✓	✗	34.6	28.5	40.8	-	-	-	24.1	27.6	35.8	14.3	18.7
#SegGPT-L [50]	COCO+ADE+VOC+... (0.25M)	Generalist	Mask	✓	✗	75.6	72.5	78.6	-	-	-	74.7	75.1	80.2	67.4	75.9
#PerSAM-L [70]	SAM+DAVIS (11M)	Generalist	Mask	✗	✓	60.3	56.6	63.9	-	-	-	*	*	*	*	*
SEEM-T	COCO+LVIS (0.12M)	Generalist	Mask/Single Point	✓	✓	60.4	57.6	63.3	62.7	58.9	66.4	51.4	55.6	44.1	59.2	46.9
SEEM-B	COCO+LVIS (0.12M)	Generalist	Mask/Single Point	✓	✓	62.8	59.5	66.2	67.2	63.6	70.9	53.8	60.0	44.5	63.5	47.2
SEEM-L	COCO+LVIS (0.12M)	Generalist	Mask/Single Point	✓	✓	58.9	55.0	62.8	62.2	58.3	66.0	50.0	57.2	38.2	61.3	43.3

Ablation study results

1. LVIS mask annotation will improve interactive segmentation results.
2. Training from scratch only hurts referring segmentation performance.
3. Increasing interactive training iterations does help for interactive segmentation.

Ablation	Fix	#Iter	Pos	Neg	COCO			Referring Segmentation			Pascal VOC		DAVIS17		
					PQ	mAP	mIoU	cIoU	mIoU	AP@50	NoC50	NoC90	JF	J	F
Baseline	Y	0	✓	✗	50.7	39.5	60.8	57.9	63.3	71.6	1.74	5.43	59.6	55.8	63.5
- LVIS	✓	2	✓	✓	51.0	39.8	62.2	58.6	63.9	72.6	1.57	4.91	59.5	55.9	63.1
+ Negative	✓	0	✓	✓	50.9	39.8	61.4	58.8	64.0	72.6	1.81	5.41	60.1	56.3	63.9
+ Scratch	✗	3	✓	✓	50.2	39.5	60.7	51.4	59.2	67.0	1.45	4.41	60.6	57.7	63.4
+ Iter	✓	1	✓	✓	50.7	39.7	60.5	58.3	63.4	71.3	1.76	5.14	59.2	55.4	63.0
	✓	2	✓	✓	50.5	39.5	61.0	58.0	63.2	71.6	1.78	5.20	59.6	56.2	63.0
	✓	3	✓	✓	50.4	39.5	61.0	58.0	63.0	71.5	1.55	4.67	59.9	56.4	63.5
	✓	5	✓	✓	50.6	39.4	60.9	58.4	63.4	71.6	1.54	4.59	59.7	56.3	63.1

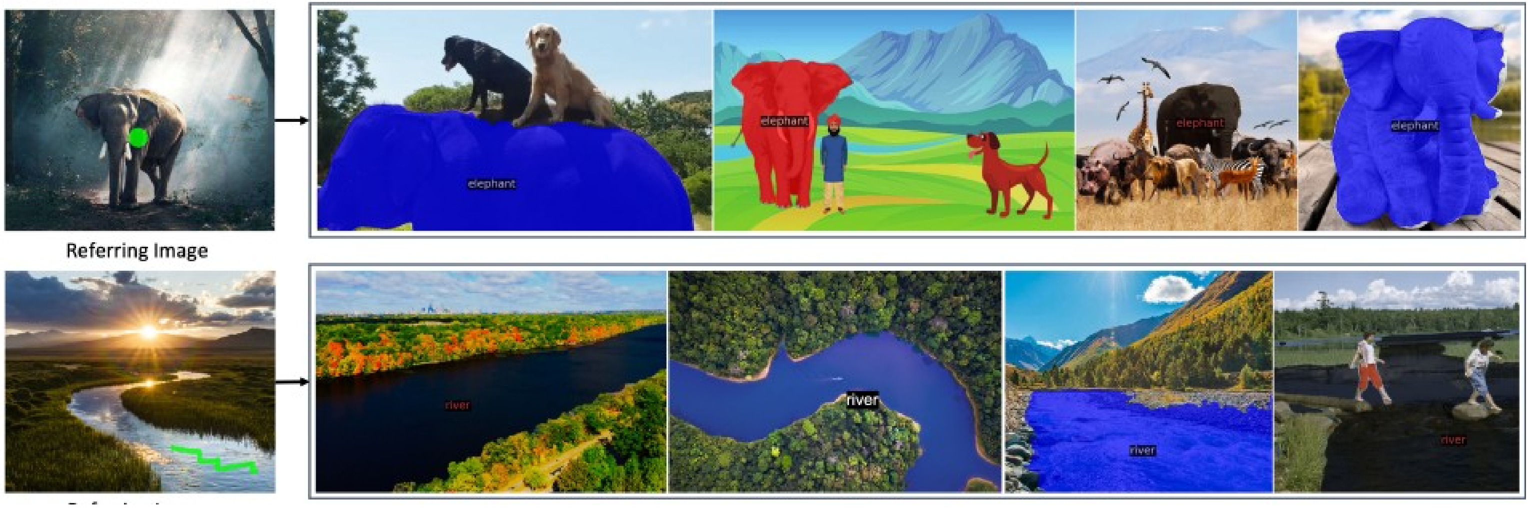
Qualitative results

Based on the proposed prompting scheme and decoder design, with the same suite of parameters, SEEM supports a wide range of visual input types.

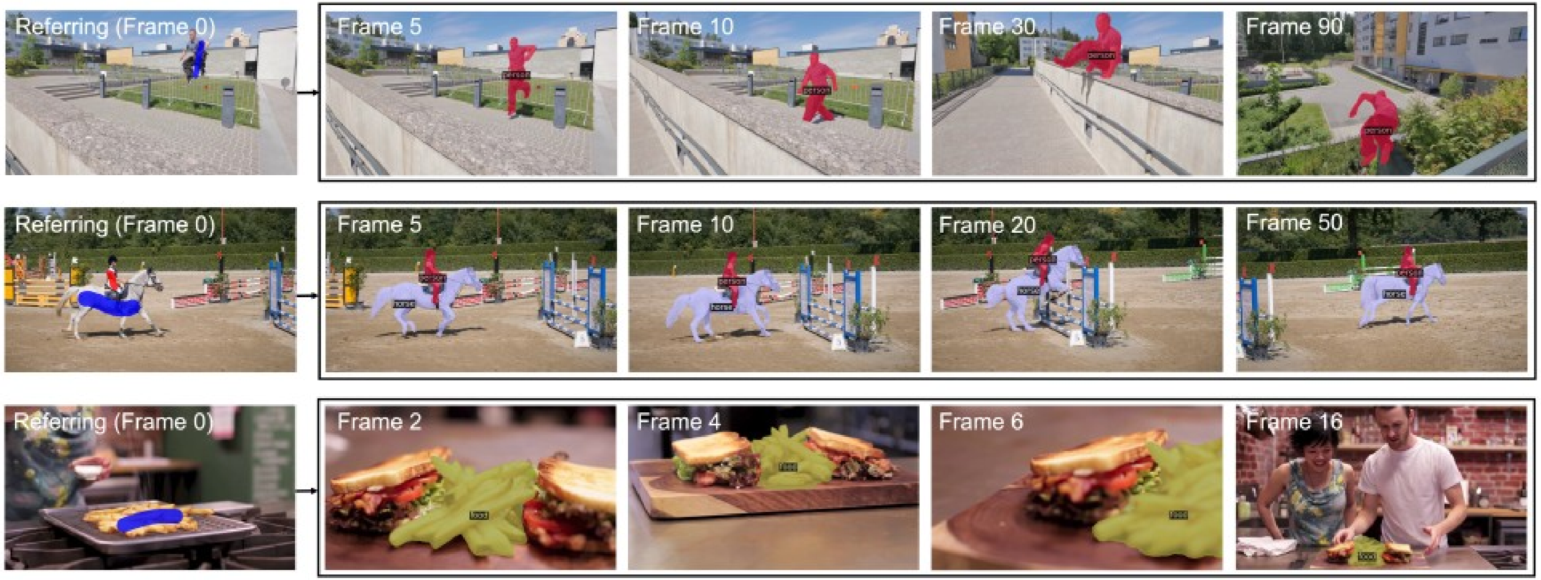
Visual prompt interactive segmentation

Text referring segmentation

Visual referring segmentation



Video object segmentation



Conclusion and future studies

SEEM yields competitive performance on several open-vocabulary and interactive segmentation benchmarks. Further studies revealed the robust generalization ability of our model in accurately segmenting images based on diverse user intents.

Quiz Time

Quiz 1

Why does interactive segmentation work although the model is not trained with any semantic labels?

Answer 1

The visual prompt features are aligned with textual features in the joint visual-semantic space. Therefore, the calculate logits are well-aligned.

Quiz 2

As shown in the paper, SEEM has bad performance in interactive segmentation. Could you modify the model or training process to improve the performance of interactive segmentation?

Answer 2

1. Use LVIS mask
2. Train from scratch.
3. Increase interactive training iterations.

Thank you