



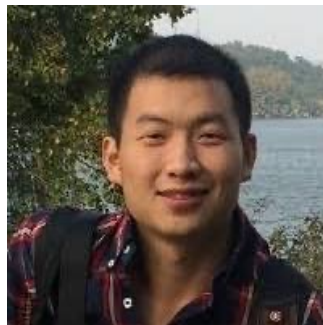
# Olympus: A Universal Task Router for Computer Vision Tasks



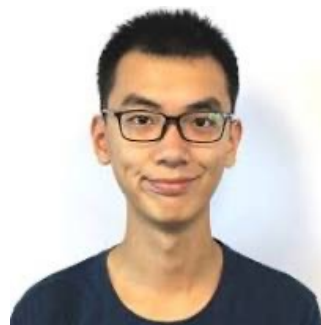
**Yuanze Lin<sup>1</sup>**



**Yunsheng Li<sup>2</sup>**



**Dongdong Chen<sup>2</sup>**



**Weijian Xu<sup>2</sup>**



**Ronald Clark<sup>1</sup>**



**Philip Torr<sup>1</sup>**













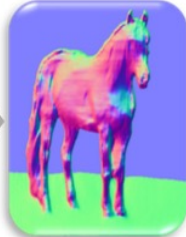

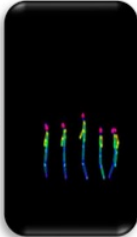







**1 - University of Oxford**

**2 - Microsoft**

# Motivation

- Integrating multiple tasks into one all-in-one foundation model:
  - (1) lead to **compromised performance** due to the conflicts between different tasks.
  - (2) **limited scalability** due to the expanding tasks across image, video and 3D domains.
  - (3) **intense computing resources** and **complicated training scheme**.
- Our Olympus method:
  - (1) handle **vision-language comprehension internally** while **delegating other tasks externally**.
  - (2) develop **task-specific routing tokens for 20 tasks** and enable MLLMs with **chain-of-action capacities**.
  - (3) collect high-quality **OlympusInstruct (446.3K)** and **OlympusBench (49.6K)** instruction datasets.

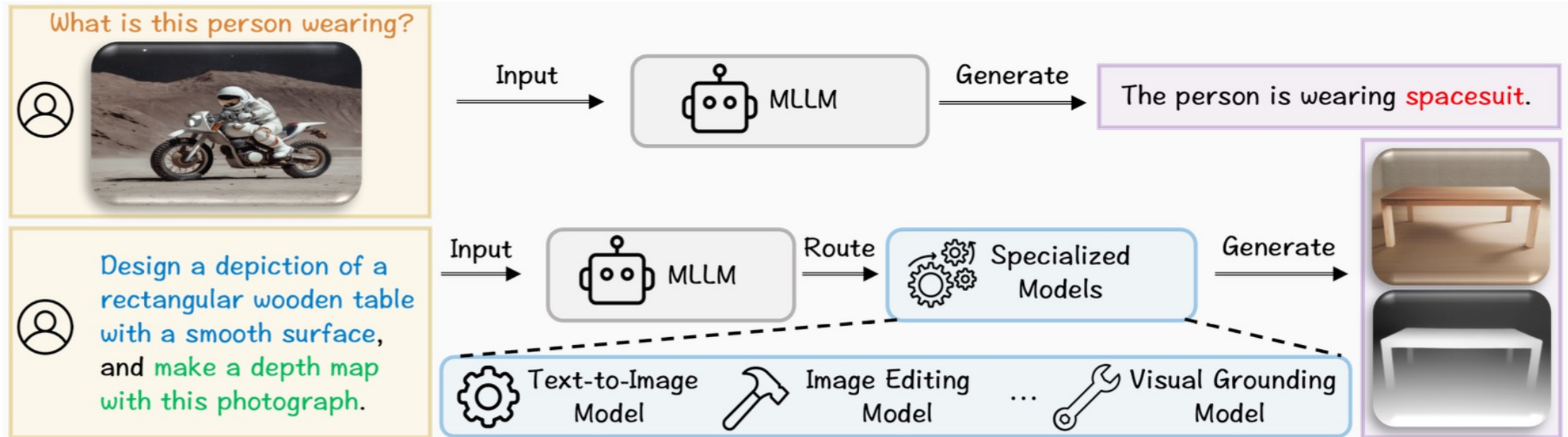
# Capacities

<b>Text to Image</b>	Craft an image displaying a chihuahua dog dressed in a vibrant, multi-colored costume.		<b>Text to Video</b>	Create a vivid video showing a fire burning within a metal container.	  
<b>Controllable Generation</b>	Reflecting the scribble imagery and the prompt 'a luxurious resort with a large swimming pool', generate an image.	 	<b>Text to 3D</b>	Form a colorful butterfly, resting on a blooming flower, delicate and intricate, 3D shape.	   
<b>Chain-of-Action</b>	Portray an intricate image of a full-body brown horse standing playfully in the grass, next, change the horse's color into red, later, could you kindly output a normal mapping result based on the given file?	  	<b>Pose Estimation</b>	Please make sure to produce a pose estimate for the input image.	 
<b>Image Editing</b>	Could you please add books near the dog? It would be interesting.	 	<b>Image to 3D</b>	Convert a 3D asset from the image in high quality.	    

# Overview

A trainable MLLM routes user prompts across specialized models:

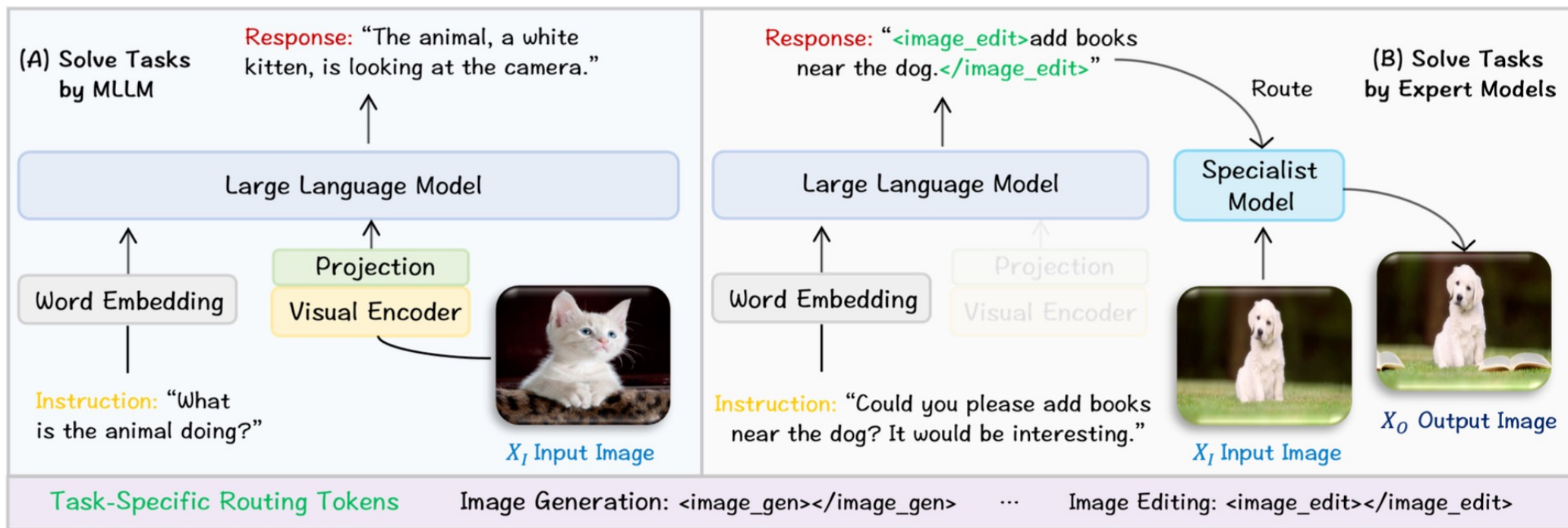
- (1) it natively handles tasks like VQA with **its inherited capacity**.
- (2) **delegate generative and vision tasks (e.g., image generation, depth estimation) to specialized models**, then aggregates the results.





# Method

It directly handles tasks like VQA using **MLLM's built-in capabilities**, while for tasks like image editing, **Olympus generates task-specific routing tokens and refined prompts to schedule specialist models**.



# Task-specific Prompt

Task-Specific Prompt
<p>Generate 50 unique user instructions with paired responses for image editing task focused on modifying objects and backgrounds. These instructions should vary significantly in language style, tone, and complexity to capture authentic interactions. Each instruction should range from short instruction into moderate and even extended instructions with multi-sentence descriptions. Occasionally use prefixes to diversify each instruction's phrasing. Responses must be extremely short, direct and practical, effectively addressing each instruction. Refer to the samples below as guidance, and come up with a wide range of prefixes and phrases to develop short, moderate and extended user instructions.</p> <p><b>Prefixes:</b></p> <p>1. Is there a way to. 2. Is it possible to. 3. I would appreciate it if you could. 4. Would you. 5. Could you. 6. Let's. 7. Please.</p> <p><b>Phrases:</b></p> <p>1. replace [object] with [new object]. 2. turn [object] into [new object]. 3. swap [object] with [new object]. 4. erase [object]. 5. let [object] [action]. 6. change [object] to [new object]. 7. remove [object]. 8. add [new object] to [object]. 9. place [new object] beside [object]. 10. make [object] [action]. 11. have [object] [action]. 12. leave [object] on [new object].</p> <p><b>User Instruction Complexities:</b></p> <p>1. <b>Short Instruction:</b> One-sentence requests using concrete objects and actions. 2. <b>Moderate Instruction:</b> Two sentences, adding context for clarity. 3. <b>Extended Instruction:</b> Over three sentences, describing emotions, effects, or scenes for richer context.</p> <p><b>User Instruction-Response Pairs:</b></p> <p><b>Short Instruction:</b> Is there a way to replace the mountain with a beach?</p> <p>- <b>Response:</b> Replace the mountain with a beach.</p> <p><b>Moderate Instruction:</b> Could you help me change dragonfly to spaceship. I believe this change will give a more inviting feel.</p> <p>- <b>Response:</b> Change dragonfly to spaceship.</p> <p><b>Extended Instruction:</b> Please replace the old fence in the background with a modern wooden one. The style should be clean and natural-looking, fitting in with a suburban backyard. I'm envisioning a style that's clean and simple, with natural tones that blend seamlessly into a cozy suburban backyard setting. The new fence should add a warm, inviting feel to the space, creating a sense of privacy without feeling closed off.</p> <p>- <b>Response:</b> Replace the old fence with a modern, clean and natural-looking wooden fence.</p>
GPT4 Output Examples
<p><b>User Instruction-Response Pairs:</b></p> <p><b>Short Instruction:</b> Could you replace the streetlight with a vintage lantern?</p> <p>- <b>Response:</b> Replace streetlight with vintage lantern.</p> <p><b>Moderate Instruction:</b> Could you add a cozy fireplace to the living room? It should create a warm and inviting atmosphere.</p> <p>- <b>Response:</b> Add a cozy fireplace to the living room.</p> <p><b>Extended Instruction:</b> I'd appreciate it if you could swap the city skyline with a mountain range. The mountains should look majestic and slightly misty, as if it's early morning. I want the scene to feel natural and peaceful, with a sense of solitude in the wilderness.</p> <p>- <b>Response:</b> Replace city skyline with misty mountains.</p>

# Task-specific Routing Tokens

Image Generation: <image\_gen>...</image\_gen>

**User Instruction:** Please craft an image displaying a chihuahua dog dressed in a vibrant, multicolored costume.

**Response:** <image\_gen>a chihuahua dog dressed in a vibrant, multicolored.</image\_gen>

# Task-specific Routing Tokens

## Chain-of-Action Setting:

**User Instruction:** In homage to the pose imagery and the prompt 'a majestic castle', generate an image. In the following step, please refine the image by adding green trees.

**Response:** <pose\_to\_image>a majestic castle</pose\_to\_image><image\_edit>adding green trees</image\_edit>.



# Training Objective

Next token prediction with the cross-entropy loss:

$$P(Y_a|\mathcal{F}_v, \mathcal{F}_t) = \prod_{i=1}^L P_{\theta}(y_i|\mathcal{F}_v, \mathcal{F}_t, Y_{a,<i}). \quad (1)$$

Here,  $L$  represents the sequence length of the response  $Y_a$ ,  $\mathcal{F}_v$  denotes the visual embedding which is adopted for those multimodal instructions, and  $\theta$  means the trainable parameters of MLLMs. The notation  $Y_{a,<i}$  denotes all tokens preceding the current token  $y_i$ , and  $\mathcal{F}_t$  represents the input instruction embeddings.

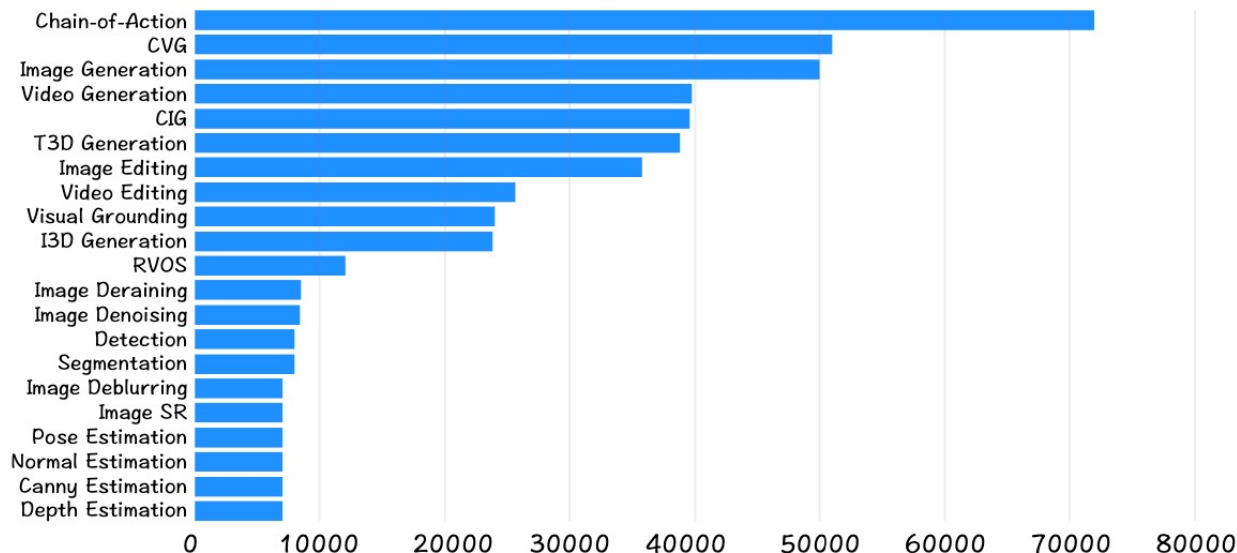
# Dataset Statistic

## Additional Covered Tasks

Image Generation, Image Editing, Controllable Image Generation (Canny, Pose, Segmentation, Depth, Normal, Scribble), Video Generation, Text-to-3D Generation, Image-to-3D Generation, Image Deblurring, Image Super-Resolution, Image Deraining, Image Denoising, Pose Estimation, Normal Estimation, Canny Estimation, Depth Estimation, Visual Grounding, Object Detection, Object Segmentation, Referring Video Object Segmentation, Controllable Video Generation (Canny, Pose, Segmentation, Depth, Normal, Scribble), Video Editing

(a) 20 covered tasks in Olympus framework.

## # of Samples Per Task

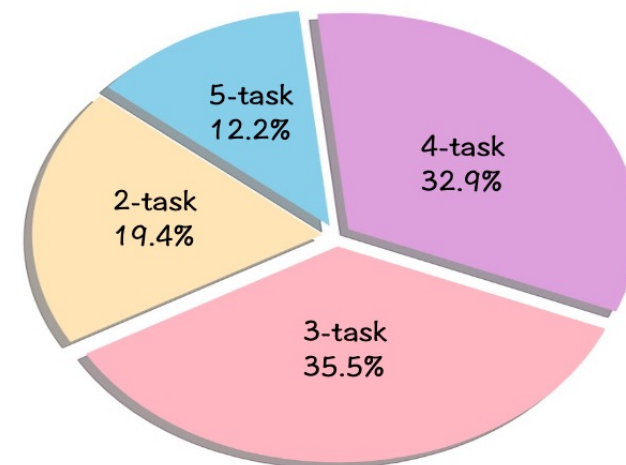


(b) Number of instructions for different tasks.

## Dataset Statistic

# of Training Instructions (Single Task)	381.5K
# of Training Instructions (Chain-of-Action)	64.8K
# of Evaluation Instructions (Single Task)	49.6K
# of Evaluation Instructions (Chain-of-Action)	7.2K
Max Instruction Word Length	372
Ave Instruction Word Length	20.2
Ave Response Word Length	10.7
Ave # of COA Tasks	3.4

(c) Statistic of the collected dataset.



# of 2-task: 13955      # of 3-task: 25565  
 # of 4-task: 23709      # of 5-task: 8771

(d) Distribution of chain-of-action instructions.

# Experiment

Method	LM	Res.	VQAv2	GQA	VisWiz	SQA <sup>I</sup>	VQA <sup>T</sup>	MME-P	MME-C	MMB	MM-Vet	POPE	MMMUS
Shikra [6]	V-13B	224	77.4	-	-	-	-	-	-	58.8	-	-	-
IDEFICS-9B [27]	L-7B	224	50.9	38.4	35.5	-	25.9	-	-	48.2	-	-	-
IDEFICS-80B [27]	L-65B	224	60.0	45.2	36.0	-	30.9	-	-	54.5	-	-	-
Qwen-VL-Chat [4]	Q-7B	448	78.2	57.5	38.9	68.2	61.5	1487.5	360.7	60.6	-	-	32.9
mPLUG-Owl2 [68]	L-7B	448	79.4	56.1	54.5	68.7	58.2	1450.2	313.2	64.5	36.2	85.8	32.1
LLaVA-1.5 [36]	V-7B	336	78.5	62.0	50.0	66.8	58.2	1510.7	316.1	64.3	30.5	85.9	32.0
MobileVLM-3B [10]	M-2.7B	336	-	59.0	-	61.2	47.5	1288.9	-	59.6	-	84.9	-
MobileVLM-v2-3B [11]	M-2.7B	336	-	61.1	-	70.0	57.5	1440.5	-	63.2	-	84.7	-
LLaVA-Phi [77]	P-2.7B	336	71.4	-	35.9	68.4	48.6	1335.1	-	59.8	28.9	85.0	-
Imp-v1 [54]	P-2.7B	384	79.5	58.6	-	70.0	59.4	1434.0	-	66.5	33.1	88.0	-
MoE-LLaVA-3.6B [31]	P-2.7B	384	79.9	62.6	43.7	70.3	57.0	1431.3	-	68.0	35.9	85.7	-
TinyLLaVA [72]	P-2.7B	384	79.9	62.0	-	69.1	59.1	1464.9	-	66.9	32.0	86.4	-
Bunny-3B [21]	P-2.7B	384	79.8	62.5	-	70.9	-	1488.8	289.3	68.6	-	86.8	33.0
Mipha-3B [76]	P-2.7B	384	81.3	63.9	45.7	70.9	56.6	1488.9	295.0	69.7	32.1	86.7	32.5
<b>Olympus (Ours)</b>	P-2.7B	384	80.5	63.9	48.2	70.7	53.4	1520.7	283.2	71.2	33.8	86.6	32.8

Performance across 11 multimodal benchmarks

# Experiment

Routing performance (single-task):

Method	Acc ↑	Pre ↑	Recall ↑	F1 ↑
HuggingGPT (GPT-4o mini)	70.14	76.51	72.14	75.46
HuggingGPT (GPT-4o)	81.35	85.54	81.55	83.56
<b>Olympus (Ours)</b>	<b>94.75</b>	<b>95.80</b>	<b>94.75</b>	<b>95.77</b>

Routing performance (chain-of-action):

Method	ED ↓	Pre ↑	Recall ↑	F1 ↑
HuggingGPT (GPT-4o mini)	0.45	65.14	48.51	53.14
HuggingGPT (GPT-4o)	0.35	75.03	60.23	61.25
<b>Olympus (Ours)</b>	<b>0.18</b>	<b>91.82</b>	<b>92.75</b>	<b>91.98</b>

Human evaluation (200 samples):

Method	Success Rate ↑
HuggingGPT (GPT-4o mini)	65.8
HuggingGPT (GPT-4o)	75.2
<b>Olympus (Ours)</b>	<b>86.5</b>

# Experiment (Ablation Study)

Varying task numbers on multimodal benchmarks:

# of Tasks	VQAv2	GQA	VisWiz	SQA <sup>I</sup>	VQA <sup>T</sup>	MME-P	MME-C	MMB	MM-Vet	POPE	MMMU
0	81.0	64.0	46.2	70.8	55.3	1498.3	293.2	70.1	32.6	86.6	32.4
5	80.5	64.2	45.6	70.9	53.5	1468.3	310.4	70.5	34.9	86.5	32.5
10	80.4	64.1	46.1	71.2	53.0	1546.7	333.9	70.2	33.8	86.2	32.9
20	80.5	63.9	48.2	70.7	53.4	1520.7	283.2	71.2	33.8	86.6	32.8

Performance remains stable when increasing training tasks!

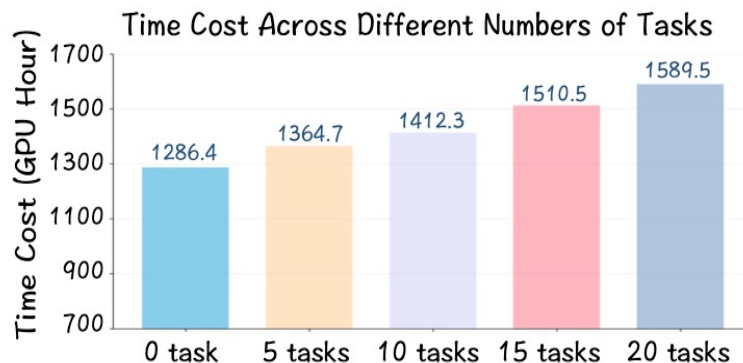
Varying task numbers for routing performance (single-task):

# of tasks	Acc ↑	Pre ↑	Recall ↑	F1 ↑
5	96.38	96.36	96.45	97.61
10	96.15	95.85	96.23	97.07
15	95.84	95.78	95.84	96.79
20	94.75	95.80	94.75	95.77

Varying task numbers for routing performance (chain-of-action):

# of tasks	ED ↓	Pre ↑	Recall ↑	F1 ↑
5	0.12	93.23	94.32	93.35
10	0.14	92.23	93.45	92.28
15	0.17	91.97	92.89	92.01
20	0.18	91.82	92.75	91.98















Increasing training tasks slightly decreases models' performance!



Only increase 23.6% time when using all the collected data!



# Applications

 <p>Identity the object in the image that is a person.</p> 	 <p>With a foundation in the segmentation picture with the idea 'a modern living room', create a new image.</p> 	<p>Generate a white Opel Insignia Car.</p>  <p>Make the car's color to be black.</p> 	<p>Envision an image of a cheerful, plush lion toy wearing a tiny crown and a cape, after that mark every object in the representation, later please generate a 3d model from the provided reference image in a lifelike manner.</p> 
 <p>Would you kindly produce a depth field from this image?</p> 	 <p>Heighten details in the entire image.</p> 	<p>Where is the car in?</p> <p>It's in a showroom.</p> <p>Can you make a canny edge detection from this photo?</p> 	 

**Thank You !!!**