CollagePrompt: A Benchmark for Budget-Friendly Visual Recognition with GPT-4V

Siyu Xu[1], Yunke Wang[1], Daochang Liu[2], Bo Du[3], Chang Xu[1]*,

[1]School of Computer Science, The University of Sydney, Australia
[2]School of Physics, Mathematics&Computing, The University of Western Australia, Australia
[3]School of Computer Science, Institute of Artificial Intelligence, Wuhan University, China.
{s.xu,c.xu,yunke.wang}@sydney.edu.au, daochang.liu@uwa.edu.au, dubo@whu.edu.cn

Abstract

Recent advancements in generative AI have suggested that by taking visual prompts, GPT-4V can demonstrate significant proficiency in visual recognition tasks. Despite its impressive capabilities, the financial cost associated with GPT-4V's inference presents a substantial barrier to its wide use. To address this challenge, we propose a budget-friendly collage prompting task that collages multiple images into a single visual prompt and makes GPT-4V perform visual recognition on several images simultaneously, thereby reducing the cost. We collect a *dataset* of various collage prompts to assess its performance in GPT-4V's visual recognition. Our evaluations reveal several key findings: 1) Recognition accuracy varies with different positions in the collage. 2) Grouping images of the same category together leads to better visual recognition results. 3) Incorrect labels often come from adjacent images. These findings highlight the importance of image arrangement within collage prompt. To this end, we construct a *benchmark* called **CollagePrompt**, which offers a platform for designing collage prompt to achieve more cost-effective visual recognition with GPT-4V. A *baseline* method derived from genetic algorithms to optimize collage layouts is proposed and two *metrics* are introduced to measure the efficiency of the optimized collage prompt. Our benchmark enables researchers to better optimize collage prompts, thus making GPT-4V more cost-effective in visual recognition. The code and data are available at this project page https://collageprompting.github.io/.

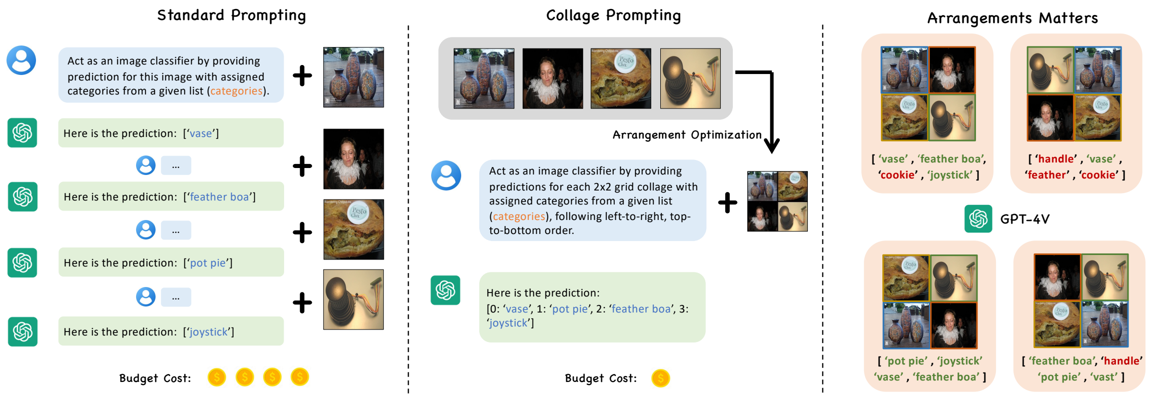
1 Introduction

With the rapid development of generative AI, various large language models (LLMs) (Chang et al., 2023; Zhao et al., 2023; Thirunavukarasu et al., 2023) have emerged as generative tools. Beyond text, these models have expanded their capabilities to include text-to-image generation such as Stable Diffusion (Rombach et al., 2022), and text-to-video generation, as seen in Sora (Brooks et al., 2024). ChatGPT (Brown et al., 2020), as the most well-known LLMs, has shown it can have natural and coherent conversations, making it a powerful tool in daily life and different industry fields. As the latest version of ChatGPT, GPT-4V is a multi-modal LLM capable of processing both text and images. This capability allows it to be applied to a wider range of applications and tasks. There are many technical reports and user studies (Li et al., 2023; Lin et al., 2023; Shi et al., 2023; Wen et al., 2023; Yang et al., 2023a; Zhou et al., 2023) about GPT-4V, which conducted thorough evaluations of its capabilities from various aspects.

In (Wu et al., 2023c), the visual capabilities of GPT-4V are investigated within the framework of zero-shot visual recognition tasks, such as image and video recognition. The evaluation of visual capabilities is quite straightforward: images and candidate categories are directly fed into GPT-4V for relevance ranking, yielding Top-1 and Top-5 prediction results. Video and point cloud data are uniformly sampled to generate a set of images, which are then processed by GPT-4V for visual recognition. GPT-4V has achieved remarkable performance across various visual recognition tasks, surpassing previous customized solutions(Wang et al., 2018b; Dosovitskiy et al., 2020). However, its financial cost associated with its inference can be significant. Specifically, performing image recognition on the ImageNet-1K dataset (Russakovsky et al., 2015) requires approximately $1 for every 20 images, leading to a total evaluation cost of over $2,500 for the entire dataset. If we consider the rate limits of maximum API requests per minute, the costs will be even higher. Thus, employing GPT-4V for visual recognition tasks is expensive, and it is meaningful to adopt a more budget-friendly way for GPT-4V's visual recognition.

[*] Corresponding authors.

Findings of the Association for Computational Linguistics: NAACL 2025, pages 6411–6433
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics



Act as an image classifier by providing prediction for this image with assigned categories from a given list (categories).

Here is the prediction: ['vase']

...

Here is the prediction: ['feather boa']

...

Here is the prediction: ['pot pie']

...

Here is the prediction: ['joystick']

Act as an image classifier by providing predictions for each 2x2 grid collage with assigned categories from a given list (categories), following left-to-right, top-to-bottom order.

Here is the prediction:
[0: 'vase', 1: 'pot pie', 2: 'feather boa', 3: 'joystick']

['vase' , 'feather boa', 'cookie' , 'joystick']

['handle' , 'vase' , 'feather' , 'cookie']

['pot pie' , 'joystick' 'vase' , 'feather boa']

['feather boa', 'handle' 'pot pie' , 'vast']

Figure 1: Visual recognition of GPT-4V with different prompting ways. (a): *Standard Prompt* takes one image as visual prompt for each GPT-4V's run. (b): *Collage Prompt* concatenates multiple images into one visual prompt and predicts class for all images in each inference. (c): The arrangement of images within collage prompting leads to significantly different results. Green indicates an accurate prediction while red indicates an wrong prediction.

In GPT-4V's visual recognition, only an image is used as the visual prompt. This standard prompting way fails to fully release the potential capacity of GPT-4V, which is able to process multiple inquiries within one prompt simultaneously. Motivated by this idea, we propose Collage Prompting, a new task of prompting for GPT-4's visual recognition for budget-friendly inference. In collage prompt, multiple images are collaged into one visual prompt with equal size. GPT-4V is then requested to recognize the class for all images within this prompt. The overall process is shown in Figure 1. Since collage prompting allows for the recognition of multiple images in one run of GPT-4V, it significantly reduces the average cost of visual recognition.

Benchmark. Based on the observation that different arrangements in the collage prompt could lead to rather large variance of accuracy in GPT-4V's recognition, we further construct a benchmark for arrangement optimization. The benchmark collects a comprehensive dataset that contains various collage prompts, which is then used to either assess the performance of collage prompts or provide a platform to develop algorithms that can optimize collage prompts for more cost-effective recognition. Based on the idea of genetic algorithm, we develop a baseline *Learn to Collage* (LCP) to optimize the arrangement. In our baseline, the collage prompt is represented as a graph, and a collage predictor is used to estimate the expected accuracy of this collage prompt. LCP is then used to search for the best arrangement in several iterations. Two new metrics are proposed to measure the cost-effectiveness of the developed algorithm.

Contributions. We make three contributions in this paper. First, we propose a budget-friendly prompting approach for GPT-4V. By involving multiple images into a single visual prompt, GPT-4V can process multiple images in one inference run, thus reducing the overall expense greatly. Second, we collect the benchmark dataset of collage prompt. The datasets contain various collage prompts from the ImageNet-1K training set and their accuracy in GPT-4V's image recognition. This dataset is meaningful for studying the effectiveness of collage prompting. Third, we propose a genetic algorithm-based optimization method for collage arrangement. This approach aims to optimize the arrangement for collage prompts and improve image recognition accuracy within GPT-4V.

2 Related Works

Exploration of GPT-4V. The state-of-the-art large multi-modal model GPT-4V was firstly launched at September 2023 and has demonstrated its strong visual capability in different fields. Early works (Wu et al., 2023d; Yang et al., 2023a) conducted a user study of GPT-4V, where operations were completed by entering prompts on a web interface for GPT-4V. Furthermore, the release of GPT-4V API in November 2023 opened up new opportunities for both academic community and industry to thoroughly evaluate GPT-4V's performance across various visual benchmarks and provide quantitative data beyond what user studies can offer. GPT-4V's capacities in multimodal medical diagnosis has been explored in (Wu et al., 2023a; Yang et al., 2023b; Deng et al., 2024), where GPT-4V can process different imaging modalities like CT and MRI in medical scene. GPT-4V can also be utilized to operate robots from providing instructions by taking multimodal input in autonomous driving (Cui et al., 2024; Han et al., 2024; Wen et al., 2023) and task planning (Wake et al., 2023; Wang et al., 2024; Hu et al., 2023). Additionally, GPT-4V has been widely used in advancing video understanding (Lin et al., 2023), conducting OCR recognition (Shi et al., 2023), acting as an intelligent web agent (Zheng et al., 2024), and dealing with each observation data (Zhang and Wang, 2024). (Wu et al., 2023c) is the first work that considers to adopt extensive quantitative analysis utilizing the established visual benchmarks. However, the evaluation of GPT-4V on visual benchmarks could lead to large expense and it is important to adopt a budget-friendly inference scheme for the evaluation of GPT-4V.

Prompt Engineering in LLMs. Prompt engineering has emerged as a crucial technique for unlocking the potential of pre-trained large language models (LLMs) and vision-language models (VLMs). The concept of prompt engineering was initially explored and popularized in the LLMs (Liu et al., 2023; Tonmoy et al., 2024; Chen et al., 2023) and VLMs (Wu et al., 2023b; Bahng et al., 2022). The most common prompting way is zero-shot prompting (Radford et al., 2019; Cheng et al., 2023), which offers a paradigm shift in leveraging large LLMs. This method significantly reduces the dependency on vast amounts of training data by employing strategically formulated prompts to steer the model towards executing new tasks. While the primary focus in the field has been on creating prompts that can release the potential capacities of LLMs, we focus on developing a prompting approach that prioritizes cost-efficiency.

3 Collage Prompting

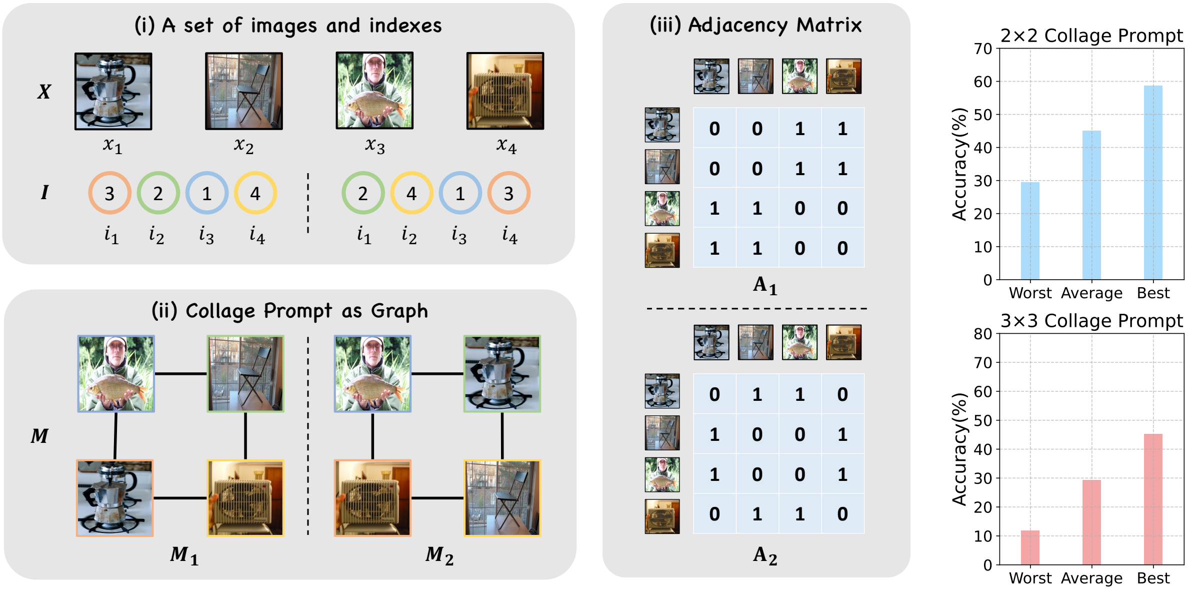
GPT-4V has enabled us to perform comprehensive visual recognition. However, each inference performed by GPT-4V incurs a financial cost, which is determined by the number and type of input and generated tokens[*]. Specifically, for image recognition tasks involving images of 512×512 resolution, approximately 5000 tokens are consumed per image. Standard prompting of GPT-4V involves presenting a single image as a visual prompt and processing each image in the dataset individually. With this standard prompting, the expense of evaluating a dataset with 10,000 images could exceed $500. This method is costly, and a more budget-friendly approach is to process multiple images simultaneously in a single inference run.

Motivated by this idea, we propose Collage Prompting, an efficient alternative to standard visual prompting. Collage Prompting involves concatenating multiple images into a single visual prompt, allowing for simultaneous processing in a single inference run. For example, employing a nine-grid collage prompt can decrease expenses to just $1/9$ of what standard individual image prompting incurs. Moreover, collage prompt not only significantly reduces costs but also processes multiple images with a single API request, thereby reducing server load and inference time. This approach is particularly beneficial for large-scale, high-frequency applications of multimodal foundational models, such as using GPT-4V for image captioning or employing GPT-4o for reading video streams.

Preliminary of collage prompt. By assembling K images into one visual prompt, the collage prompt $\mathbf{M}$ is designed to be a $\sqrt{K} \times \sqrt{K}$ grid and each grid contains one image. For example, a collage of four images might be presented in a quadrant grid, while nine images could be arranged in a nine-grid format. Supposing we have a set of K images $\mathbf{X} = [x_1, x_2, ..., x_K]$ and its related position indexes $\mathbf{I} = [i_1, i_2, ..., i_K]$, where i_j indicates the position number of image x_j in the collage prompt, starting from 0 in the top left corner to $K - 1$ in the bottom right corner. Hence, the row position r and column position c of image x_j can be specified as $r = \lfloor (i_j - 1)/\sqrt{K} \rfloor$ and $c = (i_j - 1) \mod \sqrt{K}$ respectively. While collage prompt $\mathbf{M}$ has the same size as the standard prompt, GPT-4V can thus take this collage prompt as input and generate the predicted class for all images within the collage prompt. Regarding the collage prompt $\mathbf{M}$ as graph $\mathbf{M} = (\mathbf{A}, \mathbf{F})$ with K nodes, each node $f_i \in \mathbb{R}^l$ denotes the feature of x_i. The adjacency matrix $\mathbf{A} \in \mathbb{R}^{K \times K}$ suggests the relative positions of $\mathbf{X}$, where two adjacent images are considered to have an undirected edge. For two images x_p and x_q that satisfies either $|c_p - c_q| = 1$ when $r_p = r_q$ or $|r_p - r_q| = 1$ when $c_p = c_q$, we consider these two images have an edge and set $\mathbf{A}[p, q] = \mathbf{A}[q, p] = 1$. The workflow of representing the collage prompt as a graph is illustrated in Figure 2a.

Arrangements matter. To fill multiple images into the collage prompt, the arrangement of these images could be various. As shown in Figure 2a,

[*] https://openai.com/pricing, based on pricing as of March 1, 2024.



$$A_1 = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \qquad A_2 = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

(a)

(b)

Figure 2: **(a):** The workflow of forming the collage prompt from a set of images and related indexes. For a set of images $\mathbf{X}$ with two different position indexes $\mathbf{I}$, we can obtain two collage prompts $\mathbf{M}_1$ and $\mathbf{M}_2$. Regarding $\mathbf{X}$ as the node of a graph, the adjacency matrix of $\mathbf{M}_1$ and $\mathbf{M}_2$ can be represented as $\mathbf{A}_1$ and $\mathbf{A}_2$. **(b):** The average accuracy of collage prompts within evaluation datasets using the 'Worst,' 'Average', and 'Best' arrangement.

by setting different $\mathbf{I}$, the arrangement of images within the collage prompt is different. Technically, a quadrant-grid collage has $24(4!)$ potential image arrangements, whereas a nine-grid format can exceed $360,000(9!)$ possibilities. Our findings in Figure 2b indicate that different arrangements can yield varying levels of accuracy, underscoring the importance of image arrangement. Therefore, we hope to find better arrangement within the collage to minimize the accuracy loss of GPT-4V.

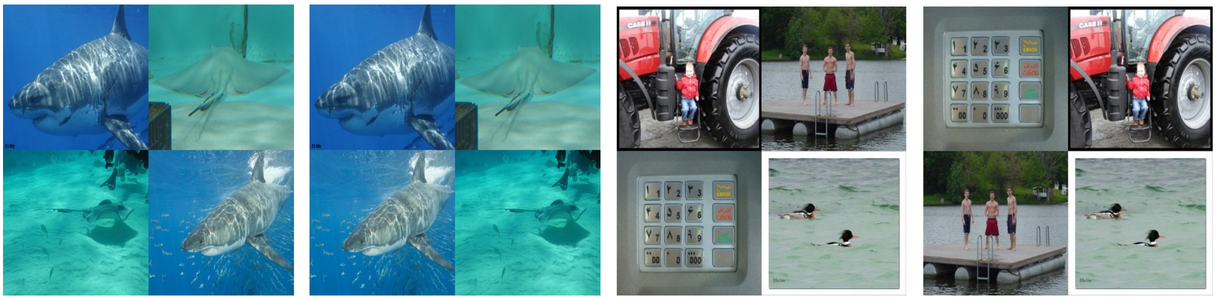
4 A Benchmark of Collage Prompting

The arrangement of images within a collage prompt significantly impacts the overall recognition accuracy of GPT-4V. For any given set of images forming a collage, there should exist one or more optimal arrangements that maximize overall recognition accuracy. Thus, we conduct a benchmark study of collage prompting with two primary objectives: **1)** to study the effect of different arrangements on GPT-4V's recognition accuracy and **2)** to provide a benchmark for developing algorithms to optimize collage arrangements. In this section, we first present a comprehensive collage prompt dataset to assess the performance of various collage prompts. Three key observations suggest that there is a need to conduct arrangement optimization. Then, we propose a baseline method to learn the layout of the collage and two metrics that reflect the cost-effective trait of GPT-4V's visual recognition are proposed.

4.1 Dataset

To construct a collage prompting dataset with various arrangements, we generate different collages (*i.e.*, $\mathbf{A}$) for the same set of images (*i.e.*, $\mathbf{X}$). We first uniformly sample a sub-dataset that contains 100,000 images from the training set of ImageNet-1K. This subset is then divided into L groups, and each group contains K images. By executing p random shuffles of the images within each group, we generated a collection of $L \times p$ unique collage prompts. We collect two collage prompting datasets with a quadrant-grid collage prompt and a nine-grid collage prompt. For the quadrant-grid collage prompt, L is set to be 25,000 and p is set to be 5. For the nine-grid collage prompt, L is set to be 11,111 and p is set to be 10. These collage prompts are then sent into GPT-4V's API for image recognition and the accuracy y of each prompt can be obtained. The final dataset $\mathcal{D}$ thus includes pairs $\{\mathbf{M}_i, y_i\}$ for each of the $L \times p$ prompts, providing a comprehensive basis for analyzing the effectiveness of different collage configurations in GPT-4V's visual recognition. We conducted an in-depth analysis of the collected collage prompting datasets and observed the following patterns.

Observation 1 (Position Accuracy Variance)



(a) "0: ‘great white shark’, 1: ‘stingray’, 2: ‘great white shark’, 3: ‘great white shark’" (b) "0: ‘great white shark’, 1: ‘stingray’, 2: ‘great white shark’, 3: ‘stingray’" (c) "0: ‘tractor’, 1: ‘automated teller machine’, 2: ‘red-breasted merganser’, 3: ‘dive’" (d) "0: ‘automated teller machine’, 1: ‘tractor’, 2: ‘dock’, 3: ‘red-breasted merganser’"

Figure 3: (a) and (b) demonstrate the effect of **category clustering**, where placing images of the same category together increases overall recognition accuracy. (c) and (d) illustrate **localization errors**, where GPT-4V predicts the correct labels but outputs them to incorrect positions in the collage.

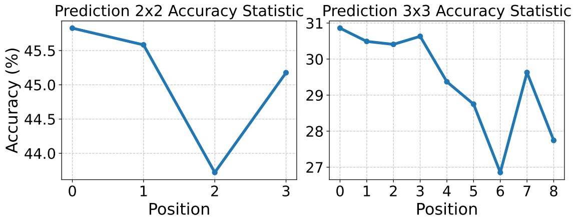


Figure 4: Average Prediction Accuracy by Position.

Different positions within the collage grid have varying accuracy in GPT-4V's visual recognition.

As shown in Figure 4, the top-left position in both 2 × 2 and 3 × 3 grids tends to have the highest accuracy, with accuracy decreasing towards the center and bottom-left positions, which have the lowest accuracy. Accuracy then improves slightly for the last row. This pattern suggests potential model fatigue when processing central images in the collage, leading to lower accuracy that recovers as the model approaches the final row. Based on this observation, a natural idea to optimize the arrangement is to place ‘hard’ images into positions with higher accuracy while leaving ‘easy’ images to remaining positions.

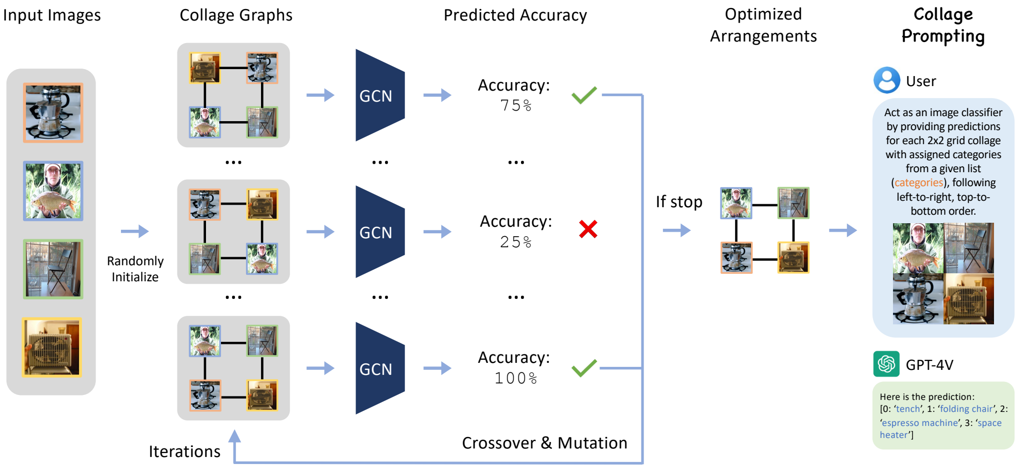
Observation 2 (Category Clustering) *Placing images of the same class together in a collage improves accuracy, while pushing images of the same class away from each other degrades the accuracy.*

We observed that in both 2 × 2 and 3 × 3 collages, placing images of the same class together significantly improves GPT-4V's overall recognition accuracy. Conversely, when the order is shuffled and images of the same class are not adjacent, the accuracy decreases. As illustrated in Figure 3a and 3b, GPT-4V predicts one of the stingrays incorrectly when the great white shark and stingray are on separate diagonals in the collage, and correctly if the same class is adjacent. This improvement can be due to clustering images of the same class reduces the complexity of batch recognition for GPT-4V.

Observation 3 (Localization Errors) *GPT-4V often makes localization errors, predicting labels for adjacent images incorrectly.*

We analyzed the prediction errors in 2 × 2 and 3 × 3 collages and found that the incorrectly predicted labels often correspond to images in adjacent positions. This indicates that the model correctly identifies the images but outputs the predictions to the wrong locations due to localization inaccuracies. For instance, in Figure 3c and 3d, GPT-4V predicts the ‘automated teller’ machine and ‘red-breasted merganser’, but outputs to the wrong positions in the collage. When the arrangement order is changed, the model outputs the correctly predicted labels to the correct positions.

Visual Bias Analysis Our analysis of collage prompting reveals that GPT-4V exhibits biases in visual recognition based on image placement. Specifically, (1) *Position Accuracy Variance* accuracy varies across collage positions, with top-left performing best and central positions showing lower accuracy, likely due to model fatigue. (2) *Category Clustering* grouping similar images enhances accuracy, while separating them reduces it. (3) *Localization Errors* labels are often misassigned to adjacent images, indicating spatial misalignment challenges. These insights highlight the need for optimized arrangements to mitigate biases. Our baseline algorithm leverages these biases to enhance collage layouts, improving recognition accuracy while maintaining cost efficiency. For further details and visual examples, see Appendix D.



Act as an image classifier by providing predictions for each 2x2 grid collage with assigned categories from a given list (categories), following left-to-right, top-to-bottom order.

Here is the prediction: [0: ‘tench’, 1: ‘folding chair’, 2: ‘espresso machine’, 3: ‘space heater’]

Figure 5: An overview of baseline method LCP. Starting with a set of images, index sets are randomly initialized, which forms multiple collage graphs. After predicting the accuracy of each collage graph via G_{θ^*}, collage graphs that achieve top-T accuracy are selected for crossover and mutation operations. This iterative process continues until reaching the maximum specified iteration and we can obtain the optimized arrangements.

4.2 Baseline: Learning to Collage

The above observations from the collage prompting dataset further highlights the importance of arrangement optimization within the collage prompt. To solve this task, we propose a baseline method Learning to Collage (LCP) in the benchmark. This baseline method involves two processes: training a collage predictor for accuracy prediction and refining collage's arrangement via genetic algorithm.

Training of Collage Predictor. As stated in the previous subsection, collage prompt $\mathbf{M}$ is represented as a graph. To predict the performance of these collage prompts, we employ GCN (Zhang et al., 2019), denoted as $G_\theta(\mathbf{A}, \mathbf{F})$, to process the graph data and predicts the expected accuracy of the collage prompt.

Given the evaluation dataset $\mathcal{D} = \{\mathbf{M}_i, y_i\}_{i=1}^{L\cdot p}$, the update of the $G_\theta(\mathbf{A}, \mathbf{F})$ at k-th iteration can be expressed as,

$$\theta_{k+1} = \theta_k - \frac{\eta}{b}\sum_{i=0}^{b} \nabla_\theta L(G_\theta(\mathbf{A}_i, \mathbf{F}_i), y_i), \tag{1}$$

where b is the batch size of training, η is the learning rate and L denotes the MSE loss. At the convergence step, θ^* will be obtained and $\mathbf{G}_{\theta^*}$ can be used to indicate the expected accuracy of the collage prompt $\mathbf{M}$.

Arrangement optimization. With the trained predictor $\mathbf{G}_{\theta^*}$, we can estimate the accuracy for various arrangements in the collage prompt, which enables the selection of the most effective arrangement to enhance recognition performance with GPT-4V. Due to the vast number of potential arrangements, it is impractical to evaluate each one to identify the optimal arrangement. To efficiently search for the best arrangement within a maximum number of iterations, we use genetic algorithm (GA) that has been widely used for non-differentiable optimization problems (Wang et al., 2018a, 2019) to achieve effectively searching. Following the idea of GA, LCP alternately evaluates the quality of arrangements in the current population and searches for the optimal collage arrangement through operations such as selection, crossover, and mutation. As shown in Figure 5, LCP consists of several key stages:

- **Initialization.** In the initialization phase, for a given set of image features $\mathbf{F}$, we randomly generate a set of position index set $\mathcal{I} = \{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_P\}$ with the related adjacency matrices $\mathcal{A} = \{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_P\}$. These matrices represent different possible arrangements of the collage prompt.

- **Evaluation.** Subsequently, for each adjacency matrix $\mathbf{A}_i$ in $\mathcal{A}$, we predict its expected accuracy using $\hat{y}_i = G_{\theta^*}(\mathbf{A}_i, \mathbf{F})$, resulting in a set of predicted accuracy $\mathcal{Y} = \{\hat{y}_0, \hat{y}_1, \ldots, \hat{y}_P\}$.

- **Selection.** During the selection phase, we choose a subset $\tilde{\mathcal{I}}$ of arrangements from $\mathcal{I}$ that correspond to the top-T accuracy in $\mathcal{Y}$, indicating the most promising arrangements, which are then preserved in the next iteration.

Table 1: Benchmark results of collage prompting with different sizes.

	Datasets	Cost($/1k)	Top-1 Accuracy		CER		PCE	
			Random	Baseline	Random	Baseline	Random	Baseline
2×2	ImageNet-1K	$12.83	39.4%	**45.7%**	4.99	**5.38**	5.49	**10.60**
	Caltech101	$1.81	88.4%	**90.8%**	13.37	**13.52**	2.15	**3.18**
	OxfordPets	$1.25	70.1%	**71.8%**	13.20	**13.34**	1.18	**1.20**
	StanfordCars	$5.28	29.8%	**32.0%**	5.65	**5.88**	1.74	**1.83**
	Flowers102	$2.15	49.8%	**51.1%**	9.75	**9.88**	1.36	**1.39**
	Food101	$2.02	62.2%	**64.2%**	11.05	**11.21**	1.40	**1.46**
	Aircraft	$2.15	**18.5%**	17.7%	**5.57**	5.42	1.45	**1.42**
	SUN397	$5.39	46.5%	**48.7%**	7.12	**7.29**	4.23	**6.02**
	DTD	$1.27	48.6%	**52.0%**	11.02	**11.40**	1.44	**1.71**
	EuroSAT	$0.86	42.9%	**53.4%**	11.21	**12.52**	**1.47**	1.16
	UCF101	$2.06	55.2%	**58.1%**	10.39	**10.65**	1.26	**1.30**
3×3	ImageNet-1K	$5.70	28.1%	**33.7%**	5.33	**5.90**	3.84	**5.01**
	Caltech101	$0.80	79.1%	**85.4%**	15.26	**15.79**	1.48	**1.89**
	OxfordPets	$0.55	53.2%	**59.5%**	13.45	**14.20**	1.12	**1.14**
	StanfordCars	$2.34	11.6%	**14.7%**	3.96	**4.68**	1.49	**1.54**
	Flowers102	$0.95	38.4%	**43.8%**	10.38	**11.12**	1.27	**1.33**
	Food101	$0.90	39.9%	**46.8%**	10.71	**11.63**	1.20	**1.24**
	Aircraft	$0.96	7.0%	**10.3%**	3.32	**4.52**	1.30	**1.35**
	SUN397	$2.39	27.6%	**36.3%**	6.89	**8.03**	1.89	**2.45**
	DTD	$0.56	37.5%	**44.1%**	11.22	**12.21**	1.23	**1.35**
	EuroSAT	$0.38	30.4%	**39.7%**	10.50	**12.14**	1.70	**2.40**
	UCF101	$0.91	37.9%	**44.0%**	10.39	**11.24**	1.18	**1.21**

Table 2: Results of GPT-4V's zero-shot visual recognition in 11 various datasets(Wu et al., 2023c).

Dataset	Cost($/1k)	Top-1 Acc.	CER
ImageNet-1K	$51.30	62.0%	4.30
Caltech101	$7.24	95.5%	9.08
OxfordPets	$4.99	92.6%	10.09
StanfordCars	$21.10	58.3%	5.26
Flowers102	$8.58	70.6%	7.52
Food101	$8.09	80.1%	8.12
Aircraft	$8.61	36.0%	5.37
SUN397	$21.55	57.7%	5.20
DTD	$5.07	59.1%	8.17
EuroSAT	$3.45	36.2%	7.19
UCF101	$8.22	81.6%	8.14

- **Crossover & Mutation.** To generate the next generation of arrangements, crossover and mutation operations are applied to the selected subset $\tilde{\mathcal{I}}$. We randomly select two arrangements from $\tilde{\mathcal{I}}$ for crossover and mutation, generating new arrangements for the next iteration. Specifically, in the crossover process, we divide two position indexes $\mathbf{I} \in \tilde{\mathcal{I}}$ into segments and cross a segment between them to generate two new position indexes. We retain the new position index with higher expected accuracy. To promote diversity, we randomly select a position index from $\tilde{\mathcal{I}}$ and mutate a randomly chosen segment of the position index. This iterative process continues, refining the search for an arrangement that maximizes the accuracy of collage prompt recognition by GPT-4V.

By iteratively employing these steps, the initial arrangements are updated efficiently until the maximum iterations are achieved. After obtaining the arrangement with the best-expected accuracy, we can apply this arrangement configuration to enhance the performance of collage prompts. Algorithm 1 outlines the steps of the LCP algorithm, while Appendix C provides more details of the baseline method. To support reproducibility, we have released the dataset, baseline code, and model weights on our project page.

4.3 Metrics

To effectively assess the performance of our collage prompting approach, we utilize **cost** (*i.e.,* $C_{n\times n}$) and **accuracy** (*i.e.,* $A_{n\times n}$) as primary evaluation metrics, where $n \times n$ denotes the size of collage prompt. Collage prompt can significantly reduce costs but often at the expense of recognition performance loss. Besides evaluating cost and accuracy separately, we introduce two new metrics for a more comprehensive analysis as follow:

Cost-Effective Ratio (CER): CER applies logarithmic transformations on both primary metrics to improve the distinction and manageability of the values. It is formulated as,

$$\text{CER} = \frac{(\log(A_{n\times n} + 1))^{\gamma}}{\log(C_{n\times n} + e)}, \quad (2)$$

Table 3: Evaluation Cost of different methods in ImageNet-1K.

	Epochs	Accuracy	Training Cost	Test Cost	Total Cost	CER
ViT-B/16	300	84.53%	\$3,876.69	\$32.77	\$3,909.46	0.022
ResNet-50	200	79.04%	\$3,063.99	\$27.64	\$2,091.63	0.038
1×1 Grid	-	62.0%	-	\$641.25	\$641.25	0.097
2×2 Grid	-	45.7%	\$9.9	\$160.37	\$170.27	0.268
3×3 Grid	-	33.7%	\$16.5	\$ 71.25	\$87.75	**0.384**

Precision-Cost Efficiency (PCE): PCE signifies the cost saved per accuracy loss, which is formulated as,

$$\mathrm{PCE} = e^{\left(\frac{|C_{n\times n} - C_{1\times 1}|}{|A_{n\times n} - A_{1\times 1}|}\right)}. \tag{3}$$

These two metrics balance the trade-offs between accuracy and cost, providing deeper insights into the efficiency of various collage configurations. We also use these metrics to evaluate the performance of our baseline algorithm.

5 Experiment

In this section, we benchmarked GPT-4V's zero-shot collage prompt recognition performance on ImageNet-1K (Russakovsky et al., 2015) and 10 other datasets (*e.g.*, Caltech101 (Fei-Fei et al., 2004), OxfordPets (Parkhi et al., 2012), StanfordCars (Krause et al., 2013), Flowers102 (Nilsback and Zisserman, 2008), Food101 (Bossard et al., 2014), Aircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), UCF101 (Soomro et al., 2012)). For the 1×1 collage prompt, we referenced the zero-shot experiment results from GPT4Vis (Wu et al., 2023c). Using the API service provided by OpenAI, we evaluated the recognition performance for 2×2 and 3×3 collage prompts. The specific model version used was "gpt-4-1106-vision-preview". We used low-resolution to input images and set a random seed to ensure deterministic results.

5.1 Benchmark Results of Collage Prompting

Our analysis reveals that utilizing Collage Prompting with GPT-4V for image recognition significantly reduces inference costs without substantially compromising accuracy as shown in Table 1. While the ImageNet-1K dataset presents greater challenges due to long text labels, leading to a more drop in accuracy, accuracy on other medium-sized datasets remains substantial. By employing different configurations of collage prompting, including larger grid sizes (2×2 and 3×3), we demonstrated a significant decrease in usage costs—approximately to 1/4 and 1/9 of the cost for single images, respectively. Despite a decrease in Top-1 accuracy as grid size increases, our baseline methods for optimizing these collage arrangements significantly reduce accuracy loss, achieving over 5% higher accuracy than random arrangements. The baseline approach highlights the balance between cost efficiency and performance preservation, making it a practical solution for leveraging large multi-modal models like GPT-4V in resource-constrained scenarios.

Our analysis demonstrates the effectiveness of collage prompting in enhancing cost-efficiency across various datasets, with $n \times n$ grid collages significantly outperforming single 1×1 images in terms of the Cost-Effective Ratio (CER) as shown in Table 1. The 3×3 grids, optimized through our collage graph optimization method, show the most notable improvements in cost efficiency, especially in datasets with simpler labels and less challenging images. Additionally, our Precision-Cost Efficiency (PCE) analysis underscores the trade-off between cost savings and accuracy loss, highlighting that our optimized 2×2 and 3×3 grid arrangements achieve substantial cost savings while minimizing accuracy loss as demonstrated in Table 1, thereby offering a balanced approach to cost-efficient image recognition with GPT-4V. Overall, these results underscore the practicality and efficiency of collage prompting in leveraging large multi-modal models for image recognition tasks under budget constraints.

5.2 Cost Analysis

We analyzed the costs associated with using AWS cloud servers for training and inference of traditional CNN and ViT models on the ImageNet-1k dataset, comparing the expenses with those of collage prompting using GPT-4V. Training ResNet-50 and ViT-B/16 from scratch incurred significant costs (\$2,091.63 and \$3,909.46, respectively). In contrast, leveraging GPT-4V for single 1×1 image prediction reduced the cost to \$641.25. Additionally, collage prompting with 2×2 or 3×3 grid configurations further decreased costs to \$170.27 and \$87.75, respectively. The Cost-Effective Ratio (CER) highlights the stark contrast between collage prompting and traditional models. For example, the 3×3 grid configuration has a CER 17 times higher than that of ViT-B/16, showcasing the cost efficiency of collage prompting.

Notably, while models like ViT and CLIP offer impressive capabilities, they entail certain barriers such as data, training, and computational resources. In contrast, GPT-4V enables zero-shot recognition with minimal setup, offering a significant advantage, especially when using collage prompting. This advantage extends to various visual recognition tasks, further emphasizing the practicality and efficiency of GPT-4V in real-world applications.

6 Limitation and Future work

Limitations. 1) *Accuracy Drop*: While collage prompting significantly reduces costs, it does trade off some recognition performance. Despite challenges with datasets like ImageNet-1K, accuracy on medium-sized datasets remains reasonable. Collage prompt is still applicable for tasks with lower accuracy requirements, such as image or video captioning. Our platform will help researchers enhance collage recognition performance, moving closer to the accuracy of standard prompting. 2) *Collage Prompt in Other LLMs*: We also tested collage prompting on other open-source and closed-source multimodal vision-language models (*e.g.*, LLAVA-1.5, Gemini 1.5 Pro) and found that these models performed poorly in visual recognition tasks. These models generated non-existent or incorrect labels, produced repetitive outputs, and failed to recognize images within the collage prompt. Appendix F provides examples of these failures.

Future Work. In this paper, we propose a budget-friendly task of collage prompting for GPT-4V's visual recognition and construct a benchmark for learning to optimize the collage prompt. Future work could explore text prompt optimization, visual prompting techniques to learn adversarial noise perturbations, LCP optimization for multiple arrangement candidates, and few-shot/many-shot methods to improve accuracy. Additionally, we will actively maintain and update CollagePrompt, expanding the baseline library, applying it to other multi-modal foundation models, and extending it to broader visual recognition tasks.

Acknowledgement

We acknowledge the National Computational Infrastructure of Australia for providing computational resources via the Sydney NCI Scheme. This work was supported in part by the Australian Research Council under Projects DP240101848 and FT230100549.

References

Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. 2022. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. 2024. Video generation models as world simulators.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2023. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2023. Unleashing the potential of prompt engineering in large language models: a comprehensive review. *arXiv preprint arXiv:2310.14735*.

Zhoujun Cheng, Jungo Kasai, and Tao Yu. 2023. Batch prompting: Efficient inference with large language model apis. *arXiv preprint arXiv:2301.08721*.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.

Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, et al. 2024. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 958–979.

Jiawen Deng, Kiyan Heybati, and Matthew Shammas-Toma. 2024. When vision meets reality: Exploring the clinical applicability of gpt-4 with vision.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Li Fei-Fei, Rob Fergus, and Pietro Perona. 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92.

Wencheng Han, Dongqian Guo, Cheng-Zhong Xu, and Jianbing Shen. 2024. Dme-driver: Integrating human decision logic and 3d scene perception in autonomous driving. *arXiv preprint arXiv:2401.03641*.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226.

Yingdong Hu, Fanqi Lin, Tong Zhang, Li Yi, and Yang Gao. 2023. Look before you leap: Unveiling the power of gpt-4v in robotic vision-language planning. *arXiv preprint arXiv:2311.17842*.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561.

Yingshu Li, Yunyi Liu, Zhanyu Wang, Xinyu Liang, Lingqiao Liu, Lei Wang, Leyang Cui, Zhaopeng Tu, Longyue Wang, and Luping Zhou. 2023. A comprehensive study of gpt-4v's multimodal capabilities in medical imaging. *medRxiv*, pages 2023–11.

Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, et al. 2023. Mmvid: Advancing video understanding with gpt-4v (ision). *arXiv preprint arXiv:2310.19773*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. 2012. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252.

Yongxin Shi, Dezhi Peng, Wenhui Liao, Zening Lin, Xinhong Chen, Chongyu Liu, Yuyi Zhang, and Lianwen Jin. 2023. Exploring ocr capabilities of gpt-4v (ision): A quantitative and in-depth evaluation. *arXiv preprint arXiv:2310.16809*.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature medicine*, 29(8):1930–1940.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Naoki Wake, Atsushi Kanehira, Kazuhiro Sasabuchi, Jun Takamatsu, and Katsushi Ikeuchi. 2023. Gpt-4v (ision) for robotics: Multimodal task planning from human demonstration. *arXiv preprint arXiv:2311.12015*.

Chaoyue Wang, Chang Xu, Xin Yao, and Dacheng Tao. 2019. Evolutionary generative adversarial networks. *IEEE Transactions on Evolutionary Computation*, 23(6):921–934.

Jiaqi Wang, Zihao Wu, Yiwei Li, Hanqi Jiang, Peng Shu, Enze Shi, Huawen Hu, Chong Ma, Yiheng Liu, Xuhui Wang, et al. 2024. Large language models for robotics: Opportunities, challenges, and perspectives. *arXiv preprint arXiv:2401.04334*.

Yunhe Wang, Chang Xu, Jiayan Qiu, Chao Xu, and Dacheng Tao. 2018a. Towards evolutionary compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2476–2485.

Yunhe Wang, Chang Xu, Chunjing Xu, Chao Xu, and Dacheng Tao. 2018b. Learning versatile filters for efficient convolutional neural networks. *Advances in Neural Information Processing Systems*, 31.

Licheng Wen, Xuemeng Yang, Daocheng Fu, Xiaofeng Wang, Pinlong Cai, Xin Li, Tao Ma, Yingxuan Li, Linran Xu, Dengke Shang, et al. 2023. On the road with gpt-4v (ision): Early explorations of visual-language model on autonomous driving. *arXiv preprint arXiv:2311.05332*.

Chaoyi Wu, Jiayu Lei, Qiaoyu Zheng, Weike Zhao, Weixiong Lin, Xiaoman Zhang, Xiao Zhou, Ziheng Zhao, Ya Zhang, Yanfeng Wang, et al. 2023a. Can gpt-4v (ision) serve medical applications? case studies on gpt-4v for multimodal medical diagnosis. *arXiv preprint arXiv:2310.09909*.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023b. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

Wenhao Wu, Huanjin Yao, Mengxi Zhang, Yuxin Song, Wanli Ouyang, and Jingdong Wang. 2023c. Gpt4vis: What can gpt-4 do for zero-shot visual recognition? *arXiv preprint arXiv:2311.15732*.

Yang Wu, Shilong Wang, Hao Yang, Tian Zheng, Hongbo Zhang, Yanyan Zhao, and Bing Qin. 2023d. An early evaluation of gpt-4v (ision). *arXiv preprint arXiv:2310.16534*.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE.

Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023a. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9(1):1.

Zhichao Yang, Zonghai Yao, Mahbuba Tasmin, Parth Vashisht, Won Seok Jang, Feiyun Ouyang, Beining Wang, Dan Berlowitz, and Hong Yu. 2023b. Performance of multimodal gpt-4v on usmle with image: Potential for imaging diagnostic support with explanations. *medRxiv*, pages 2023–10.

Chenhui Zhang and Sherrie Wang. 2024. Good at captioning, bad at counting: Benchmarking gpt-4v on earth observation data. *arXiv preprint arXiv:2401.17600*.

Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang. 2019. Hierarchical graph pooling with structure learning. *arXiv preprint arXiv:1911.05954*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024. Gpt-4v (ision) is a generalist web agent, if grounded. *arXiv preprint arXiv:2401.01614*.

Peilin Zhou, Meng Cao, You-Liang Huang, Qichen Ye, Peiyan Zhang, Junling Liu, Yueqi Xie, Yining Hua, and Jaeboum Kim. 2023. Exploring recommendation capabilities of gpt-4v (ision): A preliminary case study. *arXiv preprint arXiv:2311.04199*.

A Impact Statement

While collage prompting offers significant cost savings, especially for large-scale image recognition, it also introduces potential societal implications. The accuracy drop associated with collage prompting could have adverse effects in fields where precise recognition is paramount, such as medical imaging. These implications underscore the necessity for continued exploration into the reliability and safety of leveraging collage prompting for visual recognition tasks, particularly in fields where accuracy is critical.

A.1 Datasheet for CollagePrompt

Here, we provide a datasheet (Gebru et al., 2021) for documenting and ensuring responsible usage of the CollagePrompt Benchmark.

1. Motivation

- *For what purpose was the dataset created?* This dataset was created as a benchmark for studying the use of collage prompting to reduce the cost of GPT-4V while maintaining reasonable accuracy. It is intended for training and evaluating learning-based collage prompting optimization algorithms.

- *Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?* The dataset was created by the authors of this paper.

- *Who funded the creation of the dataset?* The creation of the dataset was funded by the Australian Research Council under Projects DP210101859 and FT230100549.

Composition

- *What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)?* The dataset comprises the original information from CollagePrompt's training and validation sets, as well as GPT-4V's prediction results for the collage prompts.

- *How many instances are there in total (of each type, if appropriate)?* This dataset includes over 110,000 2x2 and 100, 000 3x3 collage prompts with various arrangements of GPT-4V prediction results.

- *Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?* We randomly sampled 100,000 images from ImageNet to construct the 2x2 and 3x3 collage prompts.

- *What data does each instance consist of?* Each instance consists of a collage and the corresponding GPT-4V prediction results.

- *Are relationships between individual instances made explicit?* Different arrangements of collage prompts and their prediction results from the same set of images are grouped together.

- *Are there recommended data splits?* Yes, the data splits for the training and validation sets are detailed in the JSON files.

- *Are there any errors, sources of noise, or redundancies in the dataset?* The number of collage images may exceed the number of collage prompts prediction results. We have pre-cleaned the dataset to remove erroneous and unusable predictions of collage prompts.

- *Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?* The dataset requires downloading the original ImageNet-1K dataset and the downstream evaluation datasets. Using the JSON files that record the original collage information, the code can construct the collage image dataset, which is then used along with our collage prompts prediction results.

- *Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)?* No.

- *Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?* No.

Collection Process

- *How was the data associated with each instance acquired?* Each collage is constructed from the image datasets according to the order provided in the JSON file. The constructed collage, along with text prompts and dataset category labels, is then input into GPT-4V to obtain the prediction results for each sub-image in the collage.

- *What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)?* Each collage prompt is formatted and input into the GPT-4V API provided by OpenAI to obtain the corresponding prediction results. These results are then post-processed into a standardized, readable format.

- *Who was involved in the data collection process (e.g., students, crowd workers, contractors), and how were they compensated (e.g., how much were crowd workers paid)?* The data collection process did not involve any manual labor; only API usage fees were incurred.

- *Over what timeframe was the data collected?* The final version of the CollagePrompt dataset was collected in March 2024.

Uses

- *Has the dataset been used for any tasks already?* Yes, we have used this dataset to train and evaluate our baseline algorithms for optimizing collage prompts.

- *Is there a repository that links to any or all papers or systems that use the dataset?* Yes, `https://collageprompting.github.io/`.

Distribution

- *Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?* Yes, the dataset is publicly available online for anyone to access.

- *How will the dataset be distributed (e.g., tarball on website, API, GitHub)?* The dataset can be downloaded on GitHub.

- *Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?* Our dataset is distributed under CC BY 4.0. All codes on the GitHub repository are distributed under the MIT license.

- *Have any third parties imposed IP-based or other restrictions on the data associated with the instances?* No.

- *Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?* No.

Maintenance

- *Who will be supporting/hosting/maintaining the dataset?* The authors of this paper are supporting/maintaining the dataset.

- *Is there an erratum?* No.

- *Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?* Please check the dataset web page or GitHub repository for any updates.

- *If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?* Yes, they can use the code provided on our GitHub repository to generate data.

A.2 Data Hosting, Licensing, and Maintenance

The CollagePrompt Benchmark is licensed under the CC BY 4.4, and the data is hosted on Google Drive. All code used for data collection and developing baseline algorithms is distributed under the MIT license. The documentation and model checkpoints are also available on the GitHub repository. The Collage Prompt website (`https://collageprompting.github.io/`) is the central hub for all related information, including any future updates and maintenance.

B CollagePrompt Benchmark

B.1 Prompt Details

Our text prompts must include the collage filenames and category labels. After extensive experimentation, we have developed a stable version that ensures GPT-4V outputs the correct JSON format without any unrelated content. We also tested various prompting engineering techniques, but they did not significantly affect prediction accuracy. Our prompts can input single or multiple collages for prediction. When using the API, we set the batch size to 4, which does not significantly differ from predicting individual collages.

Text prompts used for 2×2 collage

```
2x2_prompt = "I want you to act as an
    Image Classifier. I will provide
    you with few 2x2 grid collages and
    a list of optional categories. Your
    task is to choose the most relevant
    category for each of the nine
    images in the grid. Start with the
    top-left image of each grid and
    proceed left to right, then down
    each row. Assign a number index
    started with 0 for each image in
    the grid. Provide the prediction in
    a dict format for each grid
    collage, key is the number index,
    and value is the most relevant
    category for each image in the
    grid. The final output is also a
    dictionary. The key is image name
    of each grid collage, and the value
    is the prediction for each grid
    collage in a dict format. Do not
    provide explanations for your
    choices or any additional
    information just the dictionary of
    predictions in a JSON format. Only
    output the predictions in one JSON
    dictionary. Here is the image([])
    and its optional categories([]).
    You have to choose strictly among
    the given categories and do not
    give any predictions that are not
    in the given category."
```

Text prompts used for 3×3 collage

```
3x3_prompt = "I want you to act as an
    Image Classifier. I will provide
    you with few 3x3 grid collages and
    a list of optional categories. Your
    task is to choose the most relevant
    category for each of the nine
    images in the grid. Start with the
    top-left image of each grid and
    proceed left to right, then down
    each row. Assign a number index
    started with 0 for each image in
    the grid. Provide the prediction in
    a dict format for each grid
    collage, key is the number index,
    and value is the most relevant
    category for each image in the
    grid. The final output is also a
    dictionary. The key is image name
    of each grid collage, and the value
    is the prediction for each grid
    collage in a dict format. Do not
    provide explanations for your
    choices or any additional
    information just the dictionary of
    predictions in a JSON format. Only
    output the predictions in one JSON
    dictionary. Here is the image([])
    and its optional categories([]).
    You have to choose strictly among
    the given categories and do not
    give any predictions that are not
    in the given category."
```

Table 4: Statistics of datasets used for evaluating collage prompting.

Datasets	Classes	Samples	Label Tokens
ImageNet-1K	1,000	50,000	4,834
Caltech101	100	2,465	428
OxfordPets	37	3,669	203
StanfordCars	106	8,041	1,814
Flowers102	102	2,463	562
Food101	101	30,300	513
FGVCAircraft	100	3,333	565
SUN397	397	19,850	1,859
DTD	47	1,692	211
EuroSAT	10	8,100	49
UCF101	101	3,783	526

B.2 Evaluation Datasets

We evaluate the performance of collage prompt on ImageNet-1K and 10 other common downstream image recognition datasets. Table 4 presents statistics regarding the number of test samples and label tokens for each dataset. Label tokens represent the number of tokens encoded by the GPT-4 tokenizer†, providing a measure of the textual labels' complexity for each dataset.

B.3 Dataset Format

Our dataset is structured as shown below. The training set begins with a random uniform sampling of images from the ImageNet-1k training dataset, followed by the construction of 2x2 and 3x3 collages according to the JSON files containing collage info in the `metainfo` directory. The evaluation datasets for all downstream datasets are formatted similarly to ImageNet-1k validation, constructing collages based on the standard data segmentation methods. All evaluation datasets consist of complete validation sets derived from ImageNet-1K.

The directory structure of CollagePrompt dataset is as follow:

```
/
└── train/
     ├── collage/Collage sets of different size
     │    ├── 2x2
     │    └── 3x3
     └── metainfo/
          ├── imagenet_train_2x2_collage_info.json   25,000
          │   collages
          └── imagenet_train_2x2_collage_pred_info.pkl
              110,250 collage prompts
```

†https://platform.openai.com/tokenizer

```
        ├── imagenet_train_3x3_collage_info.json   11,111 collages
        └── imagenet_train_3x3_collage_pred_info.pkl 102,646 collage prompts
    └── subset/
        └── img Sampled 100,000 images from ImageNet-1K
├── val/
    ├── aircraft
    ├── caltech101
    ├── dtd
    ├── eurosat
    ├── food101
    ├── imagenet/
        ├── collage/ Collage sets of different size
            ├── 2x2
            └── 3x3
        ├── metainfo/
            ├── imagenet_val_2x2_collage_info.json 12,500 collages
            └── imagenet_val_3x3_collage_info.json  5,555 collages
        └── set/
            └── img Validation set (50,000 images) of ImageNet-1K
    ├── oxflowers
    ├── oxpets
    ├── stcars
    ├── sun397
    └── ucf101
└── README.md
```

Collage Information JSON

The format of all ‘collage_info.json’ files is consistent. Each file contains the original image name, category label, and position index within the collage for each sub-image. Using this JSON file and our provided code on GitHub, users can construct collage images and evaluate the prediction results of collage prompts from GPT-4V. The contents of a collage information JSON file are shown below:

```
{
 'a72bca2a3e.jpeg': {
   '0': {'image': 'ILSVRC2012_val_00024101.JPEG',
         'synset_id': 866,
         'label': 'tractor',
         'index': 0},
   '1': {'image': 'ILSVRC2012_val_00011876.JPEG',
         'synset_id': 480,
         'label': 'automated teller machine',
         'index': 1},
   '2': {'image': 'ILSVRC2012_val_00042300.JPEG',
         'synset_id': 842,
         'label': 'swim trunks / shorts',
         'index': 2},
   '3': {'image': 'ILSVRC2012_val_00034749.JPEG',
         'synset_id': 98,
         'label': 'red-breasted merganser',
         'index': 3}},
 'd150b4fd58.jpeg': {
   '0': {'image': 'ILSVRC2012_val_00008159.JPEG',
         'synset_id': 776,
         'label': 'saxophone',
         'index': 0},
   '1': {'image': 'ILSVRC2012_val_00042315.JPEG',
         'synset_id': 123,
         'label': 'spiny lobster',
         'index': 1},
   '2': {'image': 'ILSVRC2012_val_00017726.JPEG',
...
   '3': {'image': 'ILSVRC2012_val_00016843.JPEG',
         'synset_id': 507,
         'label': 'combination lock',
         'index': 3}},
 ...
}
```

Collage Prediction JSON. The prediction results of GPT-4V for collage prompts are preprocessed and stored in JSON format for ease of use, and then saved as Pickle files to conserve storage space. Files with the suffix ‘collage_pred_info.pkl’ contain the prediction results for each collage prompt, including the original image names and their positions within the collage. Below is an example of the contents of such a file:

```
{'0b73a3623d.jpeg': {'ord': [1, 2, 0, 3],
  'pred': [0, 0, 1, 1],
  'ori': ['n07697313_12937.JPEG',
   'n03871628_38116.JPEG',
   'n04493381_61391.JPEG',
   'n02086910_6135.JPEG']},
 'eb46c53c56.jpeg': {'ord': [3, 2, 1, 0],
  'pred': [1, 1, 0, 1],
  'ori': ['n07697313_12937.JPEG',
   'n03871628_38116.JPEG',
   'n04493381_61391.JPEG',
   'n02086910_6135.JPEG']},
 '789e579a78.jpeg': {'ord': [3, 0, 2, 1],
  'pred': [1, 1, 0, 0],
  'ori': ['n07697313_12937.JPEG',
   'n03871628_38116.JPEG',
   'n04493381_61391.JPEG',
   'n02086910_6135.JPEG']},
 'f6810cfeb6.jpeg': {'ord': [0, 2, 3, 1],
  'pred': [1, 0, 1, 0],
  'ori': ['n07697313_12937.JPEG',
   'n03871628_38116.JPEG',
   'n04493381_61391.JPEG',
   'n02086910_6135.JPEG']},
 '2ed874ee56.jpeg': {'ord': [3, 0, 1, 2],
...
  'ori': ['n04335435_13775.JPEG',
   'n04023962_6300.JPEG',
   'n04507155_1825.JPEG',
   'n04447861_2306.JPEG']},
 ...
}
```

These files can be used to construct the collage image sets and to train and evaluate baseline methods for optimizing collage prompts. We also provide a complete download link for the collage image sets on GitHub, which can be used to reproduce the experimental results.

C Experimental Details

Network Details. The overall architecture of the collage predictor is depicted in Figure 6. It comprises multiple graph convolutional and pooling layers. The graph convolutional layers aggregate information from neighboring nodes, while the graph

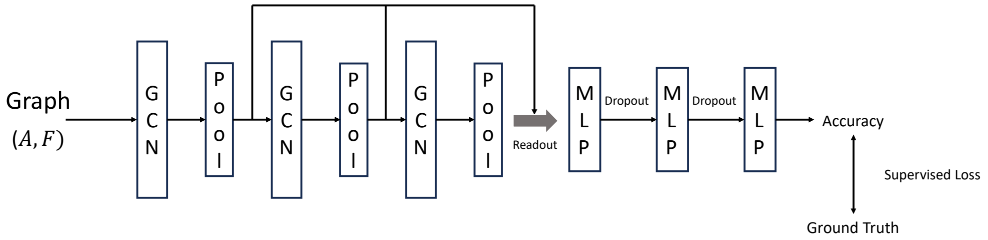


Figure 6: Network architecture of collage predictor.

pooling layers retain the sub-graph information for each node. This structure preserves the basic graph structural information and facilitates message passing. By learning graph representations hierarchically and summarizing the node representations in each layer using readout functions, the graph representations are then input into a multi-layer perception (MLP) to perform graph regression prediction tasks, specifically predicting the overall accuracy of the collage graph.

Details of Predictor Training. For training the collage predictor, both evaluation datasets for 2×2 and 3×3 collage prompt are split into training and validation sets at a 9:1 ratio, where 90% of the data is allocated for training and the remaining 10% for validation. The collage predictor is trained with a batch size of 512 and a learning rate of 0.001 for 500 epochs. We utilize the Mean Squared Error (MSE) loss function during training. The node input feature dimension of the collage graph network is set to 512. The network architecture consists of three convolutional layers and pooling layers, with a pooling ratio of 0.5. We employ the Adam optimizer to optimize the model. The dimensions of the three MLP layers are set to [256, 128, 64]. During training, we utilize an early stopping strategy to prevent overfitting. We trained the model for approximately 8 hours using an Nvidia GPU RTX 2080TI.

Details of LCP. The pseudocode for our LCP algorithm is provided in Algorithm 1. When predicting arrangements using LCP, we employ uniform crossover without allowing duplicate genes and random mutation to introduce variation in the predicted arrangements for both 2×2 and 3×3 collage prompts. For the 3×3 collage prompt, we initialize the population with 100 arrangements. In each generation, we select the top 20 arrangements with the highest accuracy to serve as parents for crossover and mutation. The evolution process continues for 10 generations, and we terminate it when the saturation threshold reaches 3. Similarly, for the 2×2 collage prompt, we begin with an initial population of 5 arrangements. We retain the top 3 arrangements in each generation based on accuracy for further reproduction. The evolution process runs for 5 generations, and we stop it when the saturation threshold also reaches 3. Finally, we evaluate the predicted best arrangements by feeding them in batches of 4 to the GPT-4V API to obtain the actual prediction accuracy.

Table 5: Top-1 accuracy, inference time and cost of collage prompts with different number of images K in GPT-4V's image recognition.

K	Top-1 Acc	Time	Cost
1×1	62.0%	8.15s	\$51.30
2×2	39.4%	2.75s	\$12.83
3×3	28.1%	1.34s	\$5.70
4×4	21.5%	1.05s	\$3.21
5×5	11.9%	0.95s	\$2.05

Details of Crossover and Mutation. During the iterative process of optimizing arrangements using LCP, crossover and mutation of arrangements are involved. The specific processes of crossover and mutation are illustrated in Figure 7. At the initial stage of each iteration, our LCP algorithm predicts the accuracy of each initial arrangement using the collage predictor and retains the top-k collage arrangements. Then, any two arrangements from the top-k are randomly selected for node crossover to obtain n partial initial node arrangements for the collages. Finally, the remaining nodes are randomly allocated (mutated) to the blank positions in the collages.

Alternative Optimization Methods. Collage optimization is a discrete problem, and our initial exploration of various methods revealed that gradient-based approaches required costly gradient estimation, making training difficult. We opted for the ge-

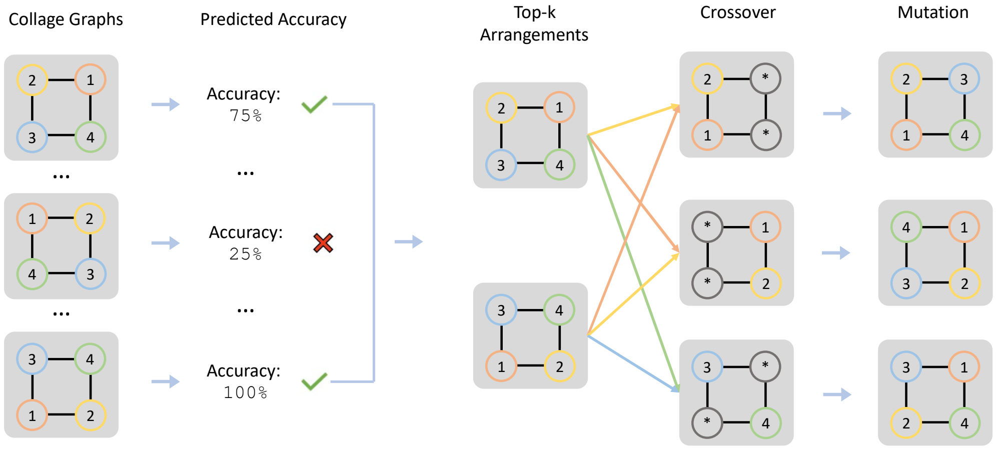
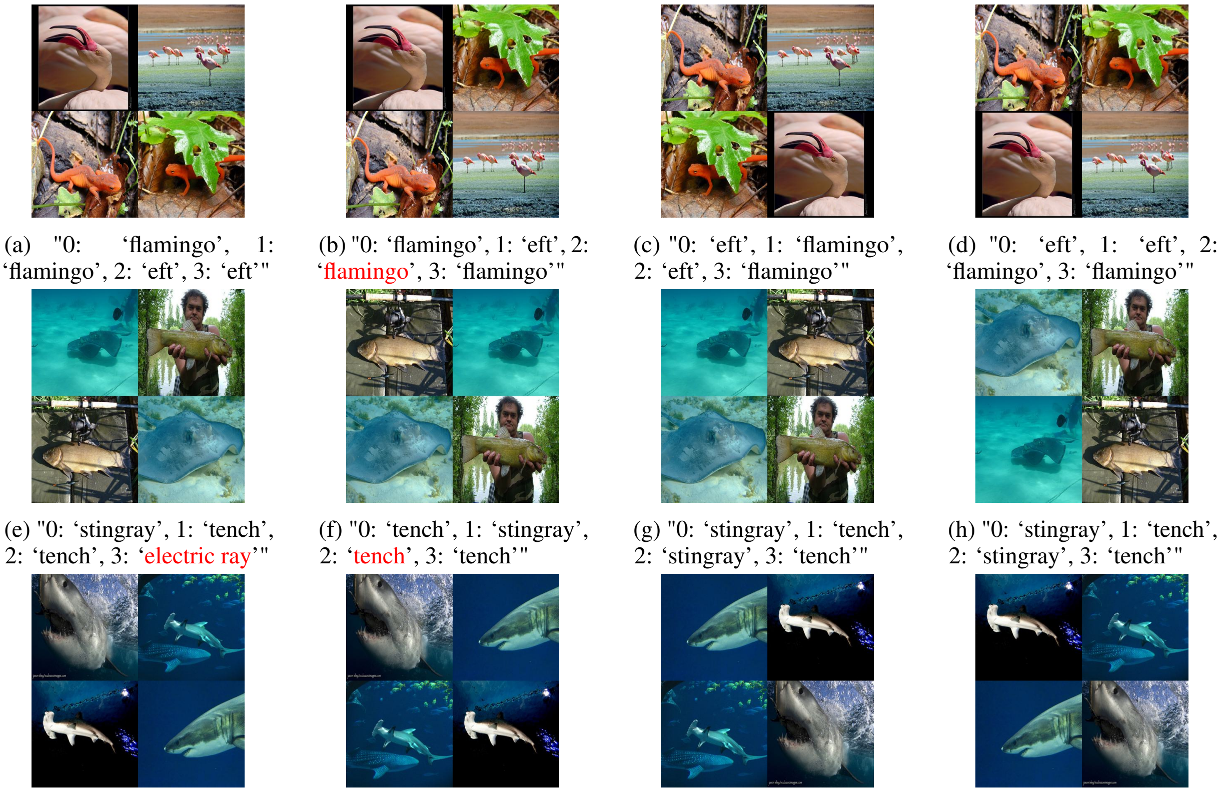


Figure 7: The process of Crossover and Mutation in the proposed LCP.



(a) "0: ‘flamingo’, 1: ‘flamingo’, 2: ‘eft’, 3: ‘eft’"

(b) "0: ‘flamingo’, 1: ‘eft’, 2: ‘flamingo’, 3: ‘flamingo’"

(c) "0: ‘eft’, 1: ‘flamingo’, 2: ‘eft’, 3: ‘flamingo’"

(d) "0: ‘eft’, 1: ‘eft’, 2: ‘flamingo’, 3: ‘flamingo’"

(e) "0: ‘stingray’, 1: ‘tench’, 2: ‘tench’, 3: ‘electric ray’"

(f) "0: ‘tench’, 1: ‘stingray’, 2: ‘tench’, 3: ‘tench’"

(g) "0: ‘stingray’, 1: ‘tench’, 2: ‘stingray’, 3: ‘tench’"

(h) "0: ‘stingray’, 1: ‘tench’, 2: ‘stingray’, 3: ‘tench’"

(i) "0: ‘great white shark’, 1: ‘great white shark’, 2: ‘hammerhead shark’, 3: ‘great white shark’"

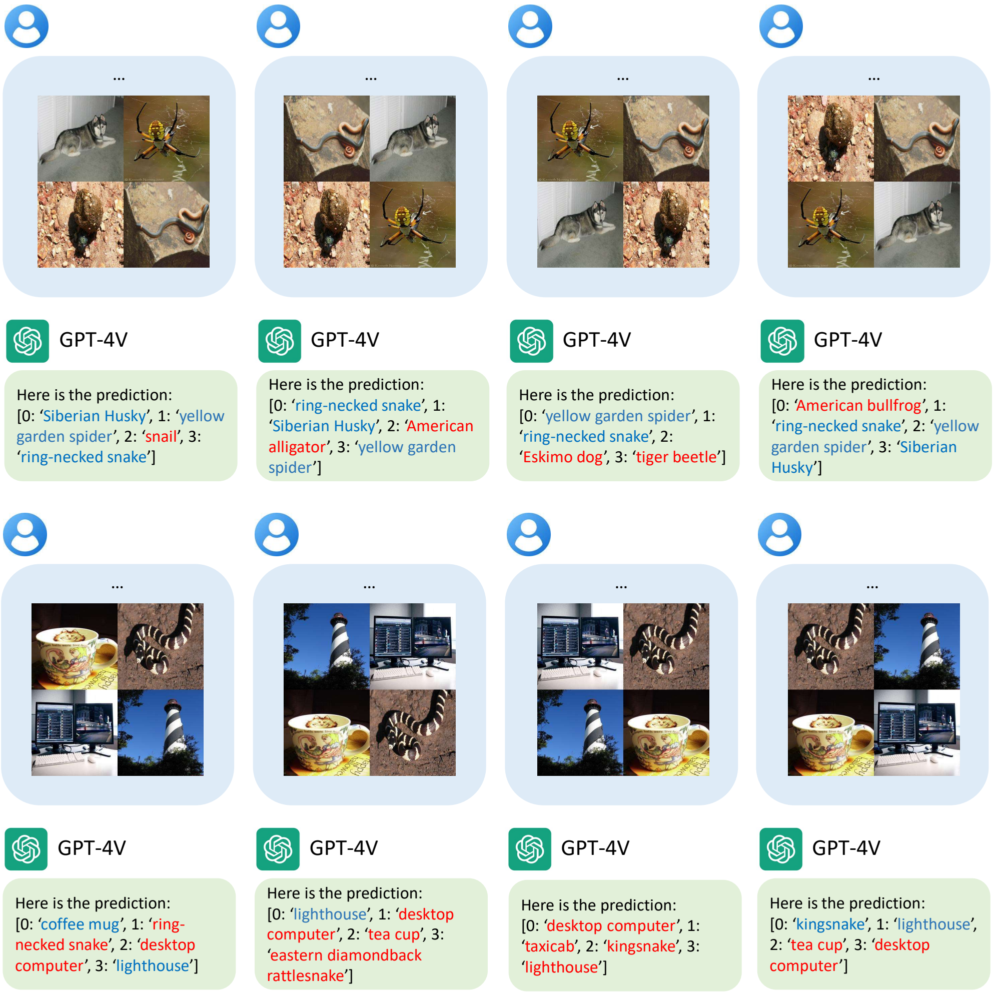
(j) "0: ‘great white shark’, 1: ‘great white shark’, 2: ‘hammerhead shark’, 3: ‘hammerhead shark’"

(k) "0: ‘great white shark’, 1: ‘great white shark’, 2: ‘great white shark’, 3: ‘great white shark’"

(l) "0: ‘hammerhead shark’, 1: ‘hammerhead shark’, 2: ‘great white shark’, 3: ‘great white shark’"

Figure 8: Examples of **Category Clustering**, showing GPT-4V’s predictions for images of the same category placed adjacently or non-adjacently.

netic algorithm due to its simplicity and efficiency in searching for optimal collage arrangements. To support further research, we provide a publicly available benchmark platform to encourage the development of advanced algorithms that enhance cost-efficiency for GPT-4V and similar models, fostering improvements in both performance and cost reduction for large-scale multimodal AI systems.



Here is the prediction:
[0: ‘Siberian Husky’, 1: ‘yellow garden spider’, 2: ‘snail’, 3: ‘ring-necked snake’]

Here is the prediction:
[0: ‘ring-necked snake’, 1: ‘Siberian Husky’, 2: ‘American alligator’, 3: ‘yellow garden spider’]

Here is the prediction:
[0: ‘yellow garden spider’, 1: ‘ring-necked snake’, 2: ‘Eskimo dog’, 3: ‘tiger beetle’]

Here is the prediction:
[0: ‘American bullfrog’, 1: ‘ring-necked snake’, 2: ‘yellow garden spider’, 3: ‘Siberian Husky’]

Here is the prediction:
[0: ‘coffee mug’, 1: ‘ring-necked snake’, 2: ‘desktop computer’, 3: ‘lighthouse’]

Here is the prediction:
[0: ‘lighthouse’, 1: ‘desktop computer’, 2: ‘tea cup’, 3: ‘eastern diamondback rattlesnake’]

Here is the prediction:
[0: ‘desktop computer’, 1: ‘taxicab’, 2: ‘kingsnake’, 3: ‘lighthouse’]

Here is the prediction:
[0: ‘kingsnake’, 1: ‘lighthouse’, 2: ‘tea cup’, 3: ‘desktop computer’]

Figure 9: Examples of **Localization Errors**: Two cases that demonstrate different arrangements within the collage prompt lead to different accuracy of classification. Blue indicates an accurate prediction while red indicates a wrong prediction.

D Visualized Results and Analysis

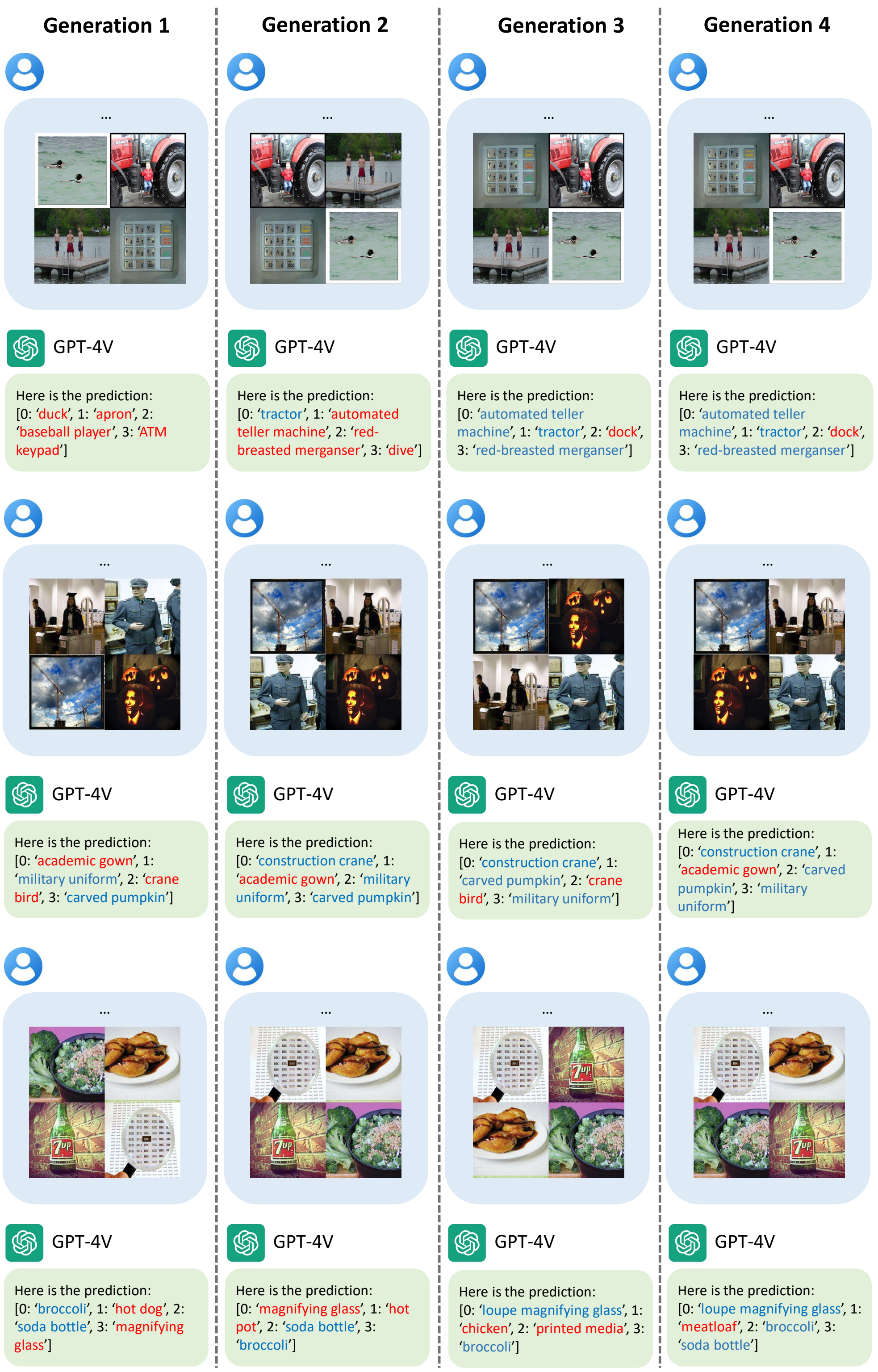
D.1 More cases about Category Clustering

As shown in Figure 8, we provide three examples of category clustering, illustrating how GPT-4V's predictions are influenced by the adjacency of images within the same category.

In the first row, we observe the behavior of GPT-4V when identifying flamingo and eft. In subfigure (a), where flamingos are grouped together and efts are grouped together, the predictions are accurate with both images correctly identified as flamingo and eft. However, in subfigure (b), when an eft is placed diagonally and not grouped with other efts, GPT-4V incorrectly predicts eft as flamingo in one instance. When flamingos and efts are grouped together again in subfigures (c) and (d), the predictions return to being correct.

The second row demonstrates the prediction tendencies for tench and stingray. In subfigure (a), when tenches and stingrays are grouped together, GPT-4V accurately predicts their respective categories. However, in subfigure (b), with tenches and stingrays positioned diagonally and not grouped together, GPT-4V incorrectly predicts ‘stingray’ as ‘tench’. This misclassification persists in subfigures (c) and (d) when the tenches and stingrays are diagonally positioned, highlighting the impact of image arrangement on GPT-4V's predictions.

In the third row, we observe the interactions between great white shark and hammerhead shark. In subfigures (a), (b), and (c), where the sharks are positioned diagonally and not grouped together, GPT-4V consistently misclassifies hammerhead shark as ‘great white shark’. However, in subfigure (d), when the sharks are grouped together, GPT-4V accurately distinguishes between the two shark species. These examples underscore the impor-



Generation 1

Here is the prediction:
[0: ‘duck’, 1: ‘apron’, 2: ‘baseball player’, 3: ‘ATM keypad’]

Here is the prediction:
[0: ‘academic gown’, 1: ‘military uniform’, 2: ‘crane bird’, 3: ‘carved pumpkin’]

Here is the prediction:
[0: ‘broccoli’, 1: ‘hot dog’, 2: ‘soda bottle’, 3: ‘magnifying glass’]

Generation 2

Here is the prediction:
[0: ‘tractor’, 1: ‘automated teller machine’, 2: ‘red-breasted merganser’, 3: ‘dive’]

Here is the prediction:
[0: ‘construction crane’, 1: ‘academic gown’, 2: ‘military uniform’, 3: ‘carved pumpkin’]

Here is the prediction:
[0: ‘magnifying glass’, 1: ‘hot pot’, 2: ‘soda bottle’, 3: ‘broccoli’]

Generation 3

Here is the prediction:
[0: ‘automated teller machine’, 1: ‘tractor’, 2: ‘dock’, 3: ‘red-breasted merganser’]

Here is the prediction:
[0: ‘construction crane’, 1: ‘carved pumpkin’, 2: ‘crane bird’, 3: ‘military uniform’]

Here is the prediction:
[0: ‘loupe magnifying glass’, 1: ‘chicken’, 2: ‘printed media’, 3: ‘broccoli’]

Generation 4

Here is the prediction:
[0: ‘automated teller machine’, 1: ‘tractor’, 2: ‘dock’, 3: ‘red-breasted merganser’]

Here is the prediction:
[0: ‘construction crane’, 1: ‘academic gown’, 2: ‘carved pumpkin’, 3: ‘military uniform’]

Here is the prediction:
[0: ‘loupe magnifying glass’, 1: ‘meatloaf’, 2: ‘broccoli’, 3: ‘soda bottle’]

Figure 10: Illustration of optimized collage arrangements and corresponding GPT-4V predictions across different generations, generated using the LCP algorithm.

tance of grouping images of the same category together in collage prompting to improve GPT-4V’s accuracy.

These observations highlight the significance of image arrangement in collage prompting. When images of similar categories are positioned adjacently, GPT-4V’s accuracy improves. Conversely, non-adjacent placement, especially diagonal position-

Algorithm 1: LCP Algorithm for Collage Arrangement Optimization

```
Parameters: $n$: size of arrangement candidates, $m$: size of selected arrangement, $\chi$: crossover rate, $\mu$: mutation rate;
Initialise generation 0:;
$k := 0$;
$P_k$ := a set of $n$ randomly-generated arrangements;
Evaluate $P_k$:;
Compute $fitness(i)$ for each $i \in P_k$;
while not stop-criterion do
    // Create generation $k + 1$:;
    // 1. Copy:;
    Select Top-$m$ arrangements from $P_k$; insert into $P_{k+1}$;
    // 2. Crossover:;
    Randomly pop out two arrangements from Top-$m$; pair them up; produce $\chi \times n$ new arrangements; insert the arrangements into $P_{k+1}$;
    // 3. Mutate:;
    Select $\mu \times n$ arrangements of $P_{k+1}$; invert a randomly-selected bit in each;
    // Evaluate $P_{k+1}$:;
    Compute fitness($i$) for each $i \in P_{k+1}$;
    // Increment:;
    $k := k + 1$;
end
return the fittest arrangement from $P_k$;
```

ing, increases the likelihood of misclassification. This underscores the need for careful consideration of image layout in tasks requiring high recognition accuracy, as the arrangement can substantially impact the performance of visual recognition models.

D.2 More cases about Localization Errors

The visualization of collage prompt reveals distinct variations in the recognition accuracy of collage images by GPT-4V across different positions within the collage. Specifically, images positioned in the top-left corner exhibit the highest recognition accuracy, while those in the bottom-left corner demonstrate the lowest accuracy. For instance, in the first row of Figure 9, the "Siberian Husky" is misclassified when positioned in the bottom-left corner but correctly identified in other positions. Moreover, relocating challenging samples to the top-left corner notably enhances GPT-4V's identification accuracy. For instance, in the second row of Figure 9, the "coffee mug", identified as a challenging sample, is correctly recognized only when placed in the top-left corner, whereas it is misclassified in other positions. Similarly, such phenomena are observed in the second row of Figure 11.

Additionally, we observed instances of mislocalization during collage image recognition by GPT-4V. This phenomenon entails the correct label of an image within the collage being predicted for the adjacent image's position. For example, in the second row of Figure 9, the "lighthouse" positioned in the bottom-left corner of the third collage is misclassified as the last image in the bottom-right corner. This mislocalization is more pronounced in the first row of Figure 11, where the "projector" is consistently misclassified as a "rotary dial telephone" when adjacent, but correctly classified as other categories when positioned diagonally. This observation offers insight into why the recognition accuracy of images in collages, particularly in datasets like EuroSAT, surpasses that of single images. When images of the same category are juxtaposed in a collage, they provide mutual cues for GPT-4V to predict the correct labels. This phenomenon was further validated through experimentation. These findings underscore the importance of considering the spatial arrangement of images within a collage when interpreting recognition accuracy and offer insights into the mechanisms underlying GPT-4V's recognition performance in such contexts.

D.3 Arrangement Optimization in LCP

Figure 10 displays various optimal collage arrangements and their corresponding predictions by GPT-4V across different generations, as generated by the LCP algorithm. In the first row examples, the "automated teller machine" was initially mispredicted in the first two generations but was correctly placed in the top-left corner in the third generation, resulting in a correct prediction by GPT-4V. The optimal arrangement remained consistent in the fourth generation, suggesting that the LCP algorithm stabilized after achieving the best arrangement.

In the second row examples, the "construction crane" was mispredicted when placed in the bottom-left corner in the first generation. However, it was correctly positioned in the top-left corner in the second generation and remained there in subsequent iterations. This indicates that the LCP

Table 6: Baseline method v.s. Brute Force Solution. 3×3 Grid Collage.

K	Method	Steps	Time	Fitness	Accuracy
3×3	LCP (Baseline)	100	1.631	0.114	0.330
		500	4.479	0.138	0.334
		1000	7.727	0.141	0.330
		1500	10.952	0.141	0.334
	Brute Force	100	0.670	0.102	0.330
		500	3.301	0.124	0.329
		1000	6.812	0.131	0.326
		1500	9.856	0.136	0.328
2×2	LCP (Baseline)	5	0.078	0.0419	0.445
		10	0.107	0.0439	0.446
		15	0.131	0.0433	0.448
		24	0.187	0.0439	0.445
	Brute Force	5	0.033	0.040	0.449
		10	0.066	0.0428	0.448
		15	0.099	0.0433	0.448
		24	0.164	0.0436	0.451

algorithm learned to place challenging samples in the top-left corner for improved prediction accuracy, while simpler samples were positioned in the bottom-left corner to enhance overall collage recognition accuracy.

In the third row examples, the "loupe magnifying glass" was initially placed in the bottom-right corner in the first generation, resulting in a misprediction by GPT-4V. Subsequently, in the second generation, the LCP algorithm positioned it in the top-left corner, still leading to a misprediction. However, in the following iterations, "loupe magnifying glass" persisted in the top-left corner, indicating the LCP predictor's confidence in this arrangement despite the initial misprediction. Eventually, in the later generations, the correct prediction was made when the "loupe magnifying glass" was placed in the top-left corner again. This example highlights the robustness of our trained LCP predictor and suggests some stochasticity in the prediction outcomes of GPT-4V.

These cases in Figure 10 further demonstrate that GPT-4V's accuracy in recognizing images within a collage varies across different positions. The LCP algorithm successfully learns the positions that yield the highest and lowest accuracy and optimally arranges the images to enhance the overall collage recognition accuracy by GPT-4V.

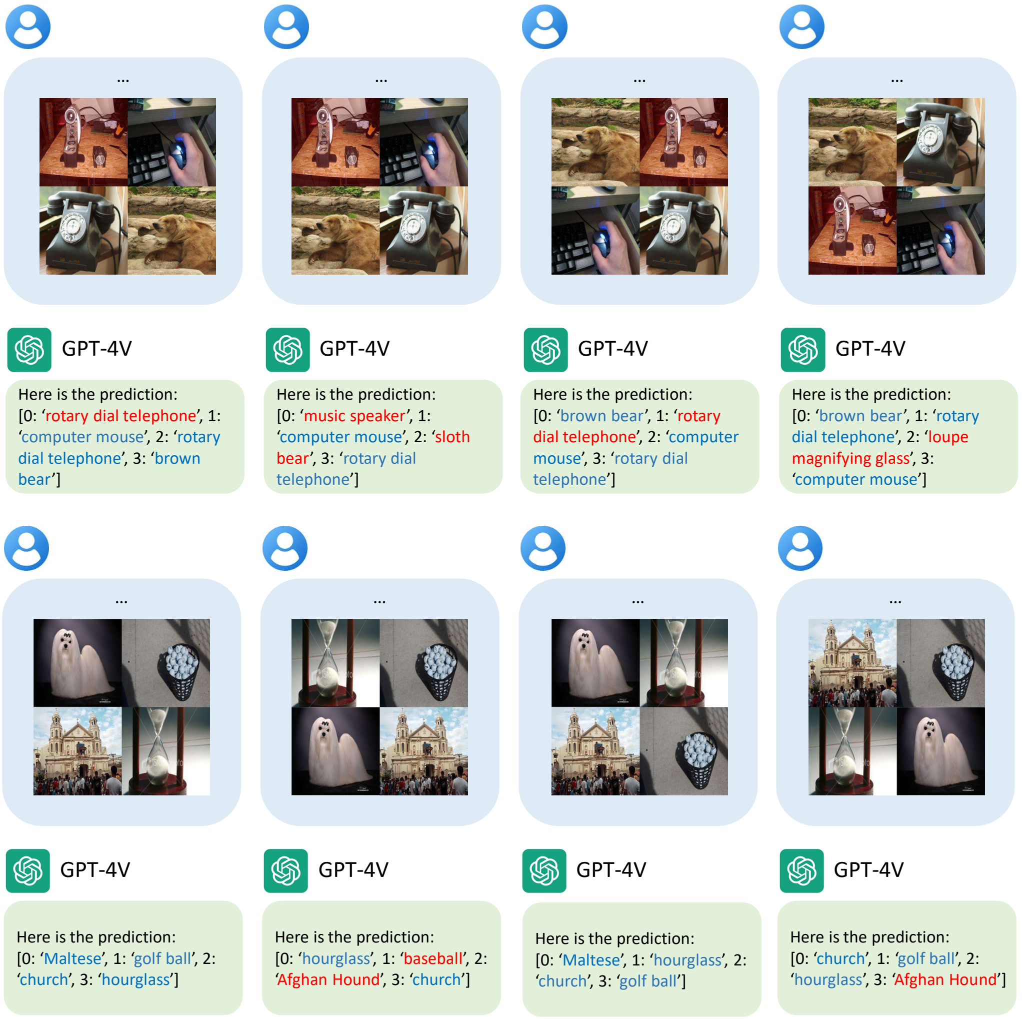
E Ablation Study

E.1 Cost-Efficiency Analysis of Collage Sizes

Table 5 illustrates the Top-1 accuracy of collage sizes ranging from 1×1 to 5×5 random grid arrangements. It also presents the inference time per image and the associated cost of using the GPT-4V API for inference per 1000 images. Notably, transitioning from single 1×1 images to 2×2 grid collages results in a reduction in accuracy of approximately 22.6%. However, the inference time and API usage cost decrease by nearly fourfold. Subsequently, each increment in grid size, from 2×2 to 5×5, leads to a decrement in accuracy by nearly 10%. Given the impracticality of using 4×4 and 5×5 grid sizes due to their significantly lower accuracy and the extensive search space for grid arrangements, focusing on optimizing the arrangement learning solely for 2×2 and 3×3 grids holds practical value. This is because 2×2 and 3×3 grids maintain acceptable accuracy levels while ensuring sufficiently low costs.

E.2 Comparison of Optimization Methods

Table 6 compares the efficacy of random initialization, brute force search using a trained model predictor, and optimization using our LCP algorithm for obtaining optimal grid arrangements for both 2×2 and 3×3 collage sizes. It is evident from



Here is the prediction:
[0: ‘rotary dial telephone’, 1: ‘computer mouse’, 2: ‘rotary dial telephone’, 3: ‘brown bear’]

Here is the prediction:
[0: ‘music speaker’, 1: ‘computer mouse’, 2: ‘sloth bear’, 3: ‘rotary dial telephone’]

Here is the prediction:
[0: ‘brown bear’, 1: ‘rotary dial telephone’, 2: ‘computer mouse’, 3: ‘rotary dial telephone’]

Here is the prediction:
[0: ‘brown bear’, 1: ‘rotary dial telephone’, 2: ‘loupe magnifying glass’, 3: ‘computer mouse’]

Here is the prediction:
[0: ‘Maltese’, 1: ‘golf ball’, 2: ‘church’, 3: ‘hourglass’]

Here is the prediction:
[0: ‘hourglass’, 1: ‘baseball’, 2: ‘Afghan Hound’, 3: ‘church’]

Here is the prediction:
[0: ‘Maltese’, 1: ‘hourglass’, 2: ‘church’, 3: ‘golf ball’]

Here is the prediction:
[0: ‘church’, 1: ‘golf ball’, 2: ‘hourglass’, 3: ‘Afghan Hound’]

Figure 11: Examples of **Localization Errors**: Two cases that demonstrate different grids within the collage prompt lead to different accuracy of classification. Blue indicates an accurate prediction while red indicates a wrong prediction.

the table that grid arrangements obtained through the model predictor-based search and optimization generally outperform those obtained through random initialization.

Moreover, the grid arrangements optimized using the LCP algorithm demonstrate higher accuracy levels while requiring fewer steps and less time compared to brute force search. For instance, the LCP algorithm for 3 × 3 grids achieves a similar accuracy as brute force search but with only 500 steps, three times fewer than brute force, and in half the time. This emphasizes the efficiency and effectiveness of using LCP algorithms for optimizing grid arrangements.

F Failure Cases from Other LVLMs

To illustrate the challenges that other large vision-language models (LVLMs) face in handling collage prompts, we present several failed response examples from different models. These outputs highlight the significant limitations of existing non-GPT-4V models in visual recognition tasks involving collage prompts.

Qwen2-VL-7B-Instruct

```
https://red-brew
concertedietet stimme in to the one of
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the the the the the the the the the
    the
```

LLAVA-v1.6-vicuna-7b

```
{
  "79f1ff56be.jpeg": {
    "airplane": 0,
    "bird": 0,
    "car": 0,
    "cat": 0,
    "deer": 0,
    "dog": 0,
    "horse": 0,
    "monkey": 0,
    "ship": 0,
    "truck": 0
  }
}
```

InternVL2-8B

```
{
  "79f1ff56be_topleft.jpg": {
    __caption_uuidf5Gddf4 vardr steht
        in the top left of the assigned
        class for this picture is the
        wrong one on the integration of
        the - at the in the of the of
        the of the of the of the of the
        of the of the of the of the of
        the of the of the of the of the
        of the of the of the of the of
        the of the
  }
}
```

Gemini Pro 1.5

```
Sorry I can't help with that image. Try
    uploading another image or
    describing the image you tried to
    upload, and I can help you that way.
```

These examples demonstrate that current non-GPT-4V models struggle significantly with collage prompts, producing outputs that are either nonsensical, repetitive, or completely uninformative. These results underscore the infeasibility of benchmarking these models against GPT-4V for collage-based visual recognition tasks. Future iterations of this work will continue evaluating LVLMs as they evolve to determine if improvements in handling collage prompts emerge.