Towards Universal and Lightweight Coverless Image Steganography with Multimodal Large Language Models Assistance

Supplementary Material

In this section, we first provide a multidimensional feature option set (MFOS) construction prompt, which is important to reproduce our results. Additionally, we discuss the reproducibility of MFOS and explain the reason for model choice in our experiments. Moreover, we supplement the data we use in experiments, including the theme word, computational complexity, generative information of critical results, the dataset, and details of steganalysis. For evaluation on construction cost and computation security, we provide a detailed explanation for the data in Tables 1 and 2 of the paper in this section as well. After that, we provide additional analysis and comparison of lightness, diversity and robustness. At last, we discuss the future work. Code is available at: https://github.com/amomiaow/MLLMStega.

1. MFOS Construction

1.1. MFOS Construction Prompt

To ensure the accuracy and robustness of secret message extraction, we represent the secret message by concrete image features in different dimensions of stego images. Therefore, we need to generate a MFOS at first. The MFOS should follow the following principles:

- Dimensions cannot conflict to ensure the harmony of the combination.
- Similar dimensions should not emerge because it may cause chaos in the sequence of extraction.
- Options of the same dimension should have a significant difference for accurate distinction.
- Since similar image content can arouse suspicion, the MFOS should ideally be diverse and controllable.

Based on the principles above, we use multimodal large language models (MLLMs) for MFOS construction. MLLMs are capable of dynamically producing rich and diverse outputs that meet input prompts. Leveraging this characteristic, we treat the potential output of MLLMs as a high-capacity and adjustable generative space, accessible through prompts. We employ prompt engineering to construct a prompt that can generate controllable MFOSs in real time. By setting different seeds, the MFOSs can have richness and diversity.

The feature set construction prompt is as follows:

```
#Role: image generation prompt engineer
You are a professional image generation pro-
mpt engineer who specializes in breaking
down themes into mutually exclusive explicit
visual dimensions.
## Rules:
1. Dimension Limitations
- The dimension range: the salient features
which can have rich options, accessories, the
pose, the background, decorative elements.
- Dimension describing color must avoid des-
cribing the overall color of the image and
instead indicate the color of specific ob-
jects/parts of the image. For example: Hair
color.
- Forbidden dimensions: Expression, Lighting,
Atmosphere.
2. Option Limitations
- There must be a very large visual difference
between options of the same dimension.
- Options between different dimensions cannot
conflict or duplicate.
3. Strict Quantity Requirements
Exactly m dimensions, each dimension have ex-
actly
n options.
## Output format:
{
"dimension_1_name": ["option1", "option2",
..., "optionn"],
"dimension_2_name": ["option1", "option2",
..., "optionn"],
// ... exactly m dimensions
}
## Additional Instructions:
- Output ONLY valid JSON without any addi-
tional text
- Use double quotes for keys and string values
- Ensure all n options per dimension are
distinct and visually diverse
```

1.2. Discussion on Reproducibility of MFOS

The reproducibility of mapping is the foundation of secret message hiding and extraction. The key lies in the reproducibility of MLLMs output. Although the same parameters and prompts can guarantee reproducibility to a certain extent, some indeterminate outputs may still occur [5]. We categorize the main patterns of MLLMs and responding usage into three types. For each type, we provide determinism enhancement strategies respectively and discuss the reproducibility.

Small-scale model in a controlled environment. For such a pattern, a simple structured output strategy is sufficient to achieve reproducibility. We generated 100 iterations of mapping using the locally run qwen2.5-VL-7B[1] model with the same parameters and prompt. These results maintained 100% consistency.

Stable middle-scale model accessing from APIs. When we

use a simple structured output strategy in this pattern at this time, some results that diverge from the majority begin to emerge. After 1000 iterations through the API of qwen2-VL-72B-Instruct with this pattern, 3.1% of the results are inconsistent with the most common outputs. Therefore, we add a strategy of generating multiple values and voting to select the most common one. For each mapping generation, we generate 10 instances. Under this strategy, the results maintained 100% consistency for 100 iterations of the complete mapping construction process.

Large-scale model accessing from APIs. Larger models possess greater generative capabilities, yielding more sampling values with similar probabilities and introducing heightened nondeterminism. Furthermore, such models typically have heavier usage. Therefore, API providers allocate them more data centers and more flexible batch balancing strategies. Consequently, ensuring reproducibility for this pattern proves exceptionally challenging. After 1000 iterations through the API of qwen3-VL-plus[11] with such a pattern, the result is different every time. We try to enhance reproducibility using the following strategy: For each dimension and each option within a dimension, generate 10 instances for each one individually and vote to select the most common value. Each time, input the previously output data as the additional prompt. Reduce the length of each dimension and the option text. After generation, sort the options in alphabetical order for each dimension. With this strategy, 90.43% consistency is achieved for 100 iterations of the complete mapping construction process.

It should be noted that the universality across different MLLMs is not the primary focus of this paper. MLLMs serve only as an auxiliary tool and does not affect the overall performance of the steganography. Therefore, we only select the API of qwen2-VL-72B-Instruct based on efficiency and stability for subsequent experiments.

2. Supplementary Description of the Data

2.1. Theme Words Used in the Experiments

Based on current trends in AIGC communities, *e.g.*, Civitai, Lexica, and Pixiv AIGC section, we construct a theme word set that closely aligns with real-world distribution. The detailed theme word and distribution can be seen in Tab. 1. The theme words used in our experiments are all chosen from Tab. 1.

2.2. Descriptions for Figure 4 of the Paper

In Figure 4 of the paper, we conduct experiments on Qwen-Image, Seedream 4.0, Midjourney V7, Nano Banana, GPT-4o, and DALL·E 3. The generation of MFOS and analysis of images are with qwen2-VL-72B-Instruct. The generated MFOS is as follows:

```
{
"Hair Style": ["Spiky", "Straight", "Curly",
"Ponytail", "Braided", "Bob Cut", "Long Loo-
se", "Short Pixie"],
"Eye Color": ["Blue", "Green", "Purple", "
Red", "Yellow", "Brown", "Pink", "Silver"],
"Outfit Style": ["School Uniform", "Ninja
Outfit", "Samurai Armor", "Maid Costume",
"Magician Robe", "Stealth Suit", "Casual
Clothes", "Futuristic Attire"],
```

Table 1. Theme words and their distribution used in our experiments. Proportion represents the proportion of images generated using the corresponding theme word among all generated images.

Category	Type	Theme word	Proportion
Characters	Female characters	woman	15%
		female	15%
	Male characters	man	5%
		male	5%
	Anime characters	anime character	20%
Scene	Landscape and architecture	landscape	2.5%
		building	2.5%
		interior design	2.5%
		traditional courtyard	2.5%
	Fantasy and science fiction	future city	2.5%
		magic world	2.5%
		alien planet	2.5%
		dreamscape	2.5%
Creative	Artistic style	pixel art	1.5%
		cartoon drawing	1.5%
		traditional Chinese painting	1%
		oil painting	1%
	Product and design	clothing	2%
		furniture	1.5%
		poster	1.5%
	Food and daily items	food	2.5%
		daily object	2.5%
Animal	Real animals	cat	1%
		dog	1%
		animal	1%
	Fantasy creatures	fantasy creature	2%

Table 2. A portion of the mapping table used in Section 4.2 of the paper. X represents an arbitrary binary bit.

Dimension	Option	Binary sequence
Eye Color	Yellow	000XXXXXXXXXXXXXXX
Eye Color	Brown	001XXXXXXXXXXXXXXX
Eye Color	Red	010XXXXXXXXXXXXXXX
...	...	...
Outfit Style	Ninja Outfit	XXX000XXXXXXXXXXX
...	...	...
Accessories	Gloves	XXXXXX100XXXXXXXXX
Accessories	Headphones	XXXXXX101XXXXXXXXX
...	...	...
Background Setting	Forest Clearing	XXXXXXXXX010XXXXXX
Background Setting	Fantasy Village	XXXXXXXXX011XXXXXX
...	...	...
Hair Style	Curly	XXXXXXXXXXXX101XXX
Hair Style	Ponytail	XXXXXXXXXXXX110XXX
...	...	...
Pose	Dancing	XXXXXXXXXXXXXXX011
Pose	Standing Confidently	XXXXXXXXXXXXXXX100
...	...	...

```
"Pose": ["Standing Confidently", "Running",
"Jumping", "Sitting Relaxed", "Crouching
Ready", "Flying", "Dancing", "Sleeping
Peacefully"],
"Background Setting": ["Urban Cityscape",
"Forest Clearing", "Castle Interior",
"Beach Sunset", "Space Station", "Fantasy
Village", "Underwater Scene", "Alien Pla-
net"],
"Accessories": ["Headphones", "Glasses",
```

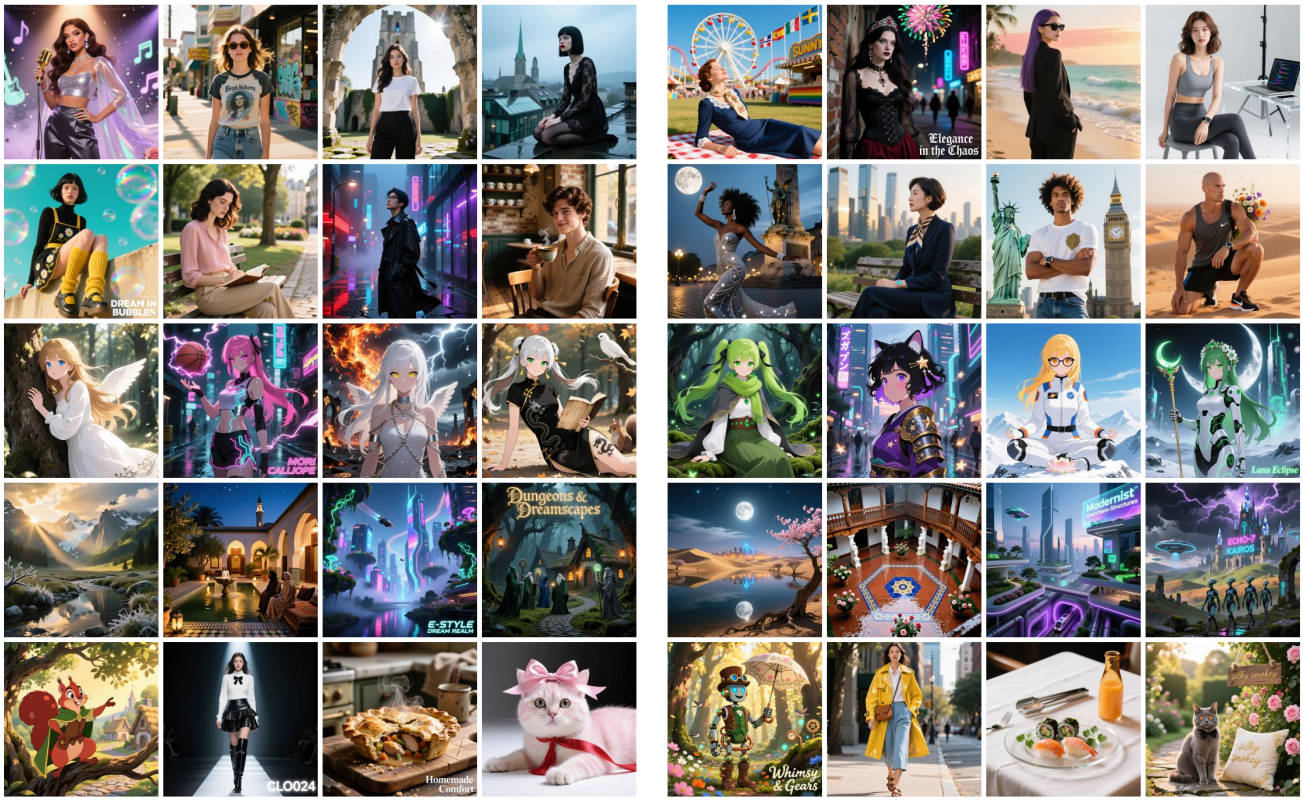

Figure 1. Clean images and stego images of our scheme in Section 4.5 of the paper. Our stego images are natural compared to normally generated images, and difficult to find regularity.

```
"Hat", "Scarf", "Necklace", "Wristwatch",
"Gloves", "Earrings"]
}
```

A portion of the mapping table is shown in Tab. 2. For each image in Figure 4 of the paper, the feature combination is {'Eye Color': 'Red', 'Outfit Style': 'Ninja Outfit', 'Accessories': 'Headphones', 'Background Setting': 'Fantasy Village', 'Hair Style': 'Ponytail', 'Pose': 'Standing Confidently'}. Prompt extending is used to enhance creativity and aesthetics.

2.3. Dataset and Details of Steganalysis

In Section 4.5 of the paper, to test the resistance to steganalysis, we first need to construct a dataset containing clean images and stego images. For our scheme, we collect 200 real-world prompts for image generation in popular communities and generate images accordingly. We consider the output images as clean images. Meanwhile, we generate 200 stego images using the image feature combinations with our scheme. Examples of clean images and stego images are shown in Fig. 1. For scheme GSIM [14], since the mapping construction is unrelated to the specific prompt, the clean images and stego images use the same prompt but with different seeds, as shown in Fig. 2. For the scheme SCTTI [6], the stego images are generated using phrases selected from a predetermined phrase database. We constructed the phrase database according to the original paper and generated stego images. Similar to the construction of clean images of our scheme, we collect real-world prompts with a similar topic to stego images and generate clean images. Since stego images generated by the fixed phrase database exhibit similar patterns as shown in Fig. 3, the steganalysis models posses a certain ability to distinguish them as indicated in Table 3 of the paper. All the steganalysis models, including XuNet [10], YeNet [12], StegNet [3], SRNet [2], and SiaStegNet [13], were trained for 100 epochs, and the best-performing epoch was used for testing. The testing result is the data in Table 3 of the paper.

3. Explanation of Tables 1 and 2 in the Paper

3.1. Storage Space Calculation

As discussed in Section 4.3 of the paper, the construction cost primarily stems from the storage space required. In this section, we conduct a detailed analysis and calculation for Table 1 of the paper. The storage cost is calculated based on the maximum information capacity declared in the original papers mentioned in Section 4.6 of the paper. For schemes that do not provide specific databases or image size parameters, we uniformly consider each image as a 128×128 RGB image, which is consistent with the previous work [6, 8]. In addition, since the previous schemes using aux-

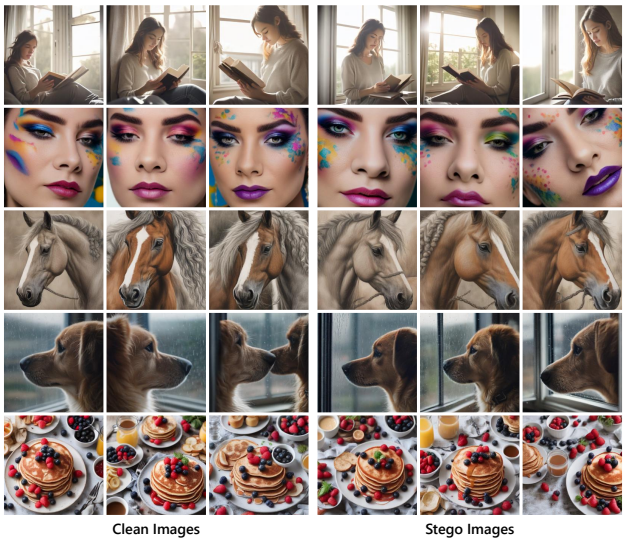

Figure 2. Clean images and stego images of GSIM [14] in Section 4.5 of the paper. The clean images and stego images are visually indistinguishable.

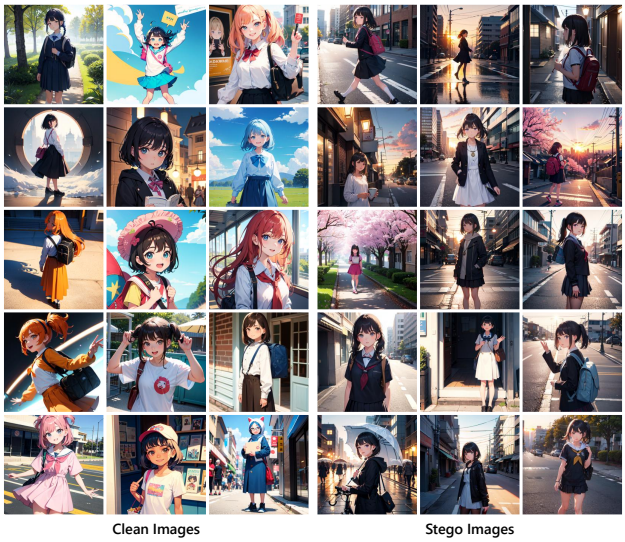

Figure 3. Clean images and stego images of SCTTI [6] in Section 4.5 of the paper. The stego images show a similar pattern, enabling the steganalysis models to achieve partial discrimination.

iliary models did not specify the size, $\mathbb{M}$ is used to represent it uniformly.

For the scheme MOR [8]: To achieve the maximum information capacity, the image databases must have 2^{24} images at least, as each image contains at most 24 bits of information. The original paper did not provide the detailed construction of databases. Therefore, we consider each image as a 128×128 RGB image. Furthermore, this scheme needs to share a same Faster RCNN model as an auxiliary model to detect objects. Therefore, the storage space they need to achieve their maximum capacity is $2^{24} \times 128 \times 128 \times 3$ Bytes $+\mathbb{M} = 768$ GB $+\mathbb{M}$ at least.

For the scheme NCIDC [15]: In this scheme, the databases include 1491 512×512 RGB images from the INRIA Holidays dataset, 2588 128×128 RGB images from the ImageNet dataset, 903 128×128 RGB images from the Caltech-256 dataset, and 120000 256×256 RGB images from the NUS-WIDE dataset. Furthermore, this scheme used a DCMH-CNN as the feature extractor. Therefore, the storage space is $1491 \times 512 \times 512 \times 3 + 2588 \times 128 \times 128 \times 3 + 903 \times 128 \times 128 \times 3 + 120000 \times 256 \times 256 \times 3$ Bytes $+\mathbb{M} \approx 23.75$ GB $+\mathbb{M}$.

For the scheme CI-CIS [7]: In this scheme, the databases includes $2520 + 500$ 512×512 RGB images from INRIA Holidays, $2520 + 500$ 128×128 RGB images from ImageNet, $3198 + 500$ 128×128 RGB images from Caltech-101, and $5708 + 500 + 800$ 128×128 RGB images from Caltech-256 to achieve a information capacity of 8 bits per image. However, to achieve the maximum information capacity declared in the original paper, the scheme requires at least 2^{15} images, according to their discussion. Based on the average image size in the original experimental setting, the storage space for the images needs to be $\big((2520 + 500) \times 512 \times 512 \times 3 + (2520 + 500 + 3198 + 500 + 5708 + 500 + 800) \times 128 \times 128 \times 3\big) \div \big(2520 + 500 + 2520 + 500 + 3198 + 500 + 5708 + 500 + 800\big) \times 2^{15}$ Bytes ≈ 5.56 GB at least. Furthermore, this scheme used a pre-training DenseNet121 network as the feature extractor. Therefore, the storage space is 5.56 GB $+\mathbb{M}$.

For the scheme ETEHG [9]: This scheme needs 2^{15} images to achieve the maximum information capacity. All of the images are resized to 256×256. In this scheme, an end-to-end hash generation model is needed to generate the hash sequence for each image. Therefore, the storage space is $2^{15} \times 256 \times 256 \times 3$ Bytes $+\mathbb{M} = 6$ GB $+\mathbb{M}$.

For the scheme GSIM [14]: Although this scheme does not need to transmit the database, it still needs to generate and store a database containing 2^{24} 512×512 RGB images to achieve the maximum information capacity. SDXL Turbo is needed to create such an image database. Therefore, the storage space is $2^{24} \times 512 \times 512 \times 3$ Bytes $+\mathbb{M} = 12288$ GB $+\mathbb{M}$.

For the scheme APAS [4]: This scheme uses the MS COCO 2014 dataset to achieve the maximum information capacity. The dataset contains 82783 images, with a size of 13.5 GB. In this scheme, no auxiliary model is needed.

For the scheme SCTTI [6]: This scheme uses 75 phrases of one theme for information hiding and image generation. According to the original paper, the storage space is 1.33 KB.

For our scheme: Our scheme only needs to store one MFOS prompt for mapping construction of any theme. Through a combination of themes, our scheme can be flexibly expanded to increase capacity. The feature set construction prompt can be seen in Sec. 1.1, which is 1.20 KB in size.

3.2. Mapping Space Size Calculation

As discussed in Section 4.5 of the paper, the computational security depends on the total number of possible mappings when the algorithm, parameters, model, and databases are publicly known. A larger mapping space size indicates that more distinct mappings can be formed, leading to higher computational security to resist exhaustion. The mapping space size calculation is based on the single-image level to more fairly evaluate the computational secu-

rity within a mapping cycle. The detailed analysis and calculation process is as follows:

For scheme NCIDC [15], ETEHG [9], APAS [4]: These schemes do not have any randomness enhancement strategies, relying on the confidentiality of construction elements to achieve security. This assumption of security is unreasonable, as it faces a serious leakage risk and does not meet the requirements of modern security design. Within the same algorithm, parameters, model, and databases, the mapping is unique for each image. Therefore, the mapping space size is 1.

For the scheme MOR [8]: In this scheme, the specific mapping depends on a key, which is defined as the number of objects in each image. For each image, there is a one-to-many relationship between images and secret messages. The number of potential mapping for each image can be represented as $\min\{N_m, N_t\}$, where N_m represents the max number of objects can be detected in the image and N_t represents the threshold number of objects in the setting. The N_m is unknown to us, as the paper did not provide the exact images used in the experiments. The value of N_t is typically 4 in most experiments, as mentioned in the experimental setting. Therefore, the mapping space size is < 4.

For scheme CI-CIS [7]: In this scheme, the specific mapping depends on the seed of the k-means++ algorithm and category number K. There are generally 2^{31} possible values for seed, as it should be a non-negative integer. In the experiments, the K is fixed for each database. Therefore, the mapping space size of this scheme is $2^{31} \approx 2.14 \times 10^9$.

For the scheme GSIM [14]: In this scheme, the specific mapping depends on the fractional parameter t, the initial random seed of SDXL Turbo, and the random seed step. In the experiments of this work, the possible values of k range from 0.3 to 5, discretized with a step size of 0.1, resulting in a total of 48 values. For the initial random seed of SDXL Turbo and the random seed step, both have 2^{31} possible values, as it should be a non-negative integer. Therefore, the mapping space size of this scheme is $48 \times 2^{31} \times 2^{31} \approx 2.21 \times 10^{20}$.

For the scheme SCTTI [6]: In this scheme, the specific mapping depends on the construction of the basic encoded mapping and randomness enhancement process. There are $\binom{N_p}{2^L}$ possible results for the construction of the basic encoded mapping, as the process involves choosing 2^L texts from N_p texts and sequentially mapping them into L-bit secret bit sequences. In this scheme, $L = 6$ and $N_p = 74$, resulting in $\binom{N_p}{2^L} = 7.18 \times 10^{11}$. The randomness enhancement process adds confusion phrases into the basic phrases for each image, resulting in a many-to-one relationship between images and secret messages. The number of potential mappings for each secret message has been calculated in this paper, which is $\frac{1}{7.4732 \times 10^{-9}} = 1.34 \times 10^8$. Therefore, the overall mapping space size is $7.18 \times 10^{11} \times 1.34 \times 10^8 = 9.62 \times 10^{19}$.

For our scheme: As in Section 4.5 of the paper, the mapping space of our scheme is $2^{31} \times m! \times (n!)^m$. In most common basic schemes, m equals 6 and n equals 8, resulting in the mapping space with a size of $2^{31} \times 6! \times (8!)^6 \approx 6.64 \times 10^{39}$.

4. Supplement for lightness

In the paper, we mainly discuss the lightness in terms of construction cost, as the primary sources of complexity in typical MCIS are the construction, transmission, and maintenance of the database and mappings. By using MLLMs to construct mappings and accessing MLLMs through APIs, we achieve the best lightness on average. Additionally, the overall computational complexity of our scheme primarily stems from three parts: the process involving MLLM access for mapping construction, the confusion algorithms used for semantic feature mapping, and the generation-validation iteration process. As discussed in Sec. 1.2, during one mapping construction, the determinism enhancement strategy requires accessing the MLLMs 10 times to ensure stable and reproducible outputs. Therefore, the complexity of this process is $O(10 \cdot C_{MLLM}) = O(C_{MLLM})$, where C_{MLLM} represents the average cost of one MLLM access through the API. For a MFOS with M dimensions and n options per dimension, the complexity of the confusion algorithm is $O(m) + O(n) = O(m+n)$. The generation–validation process iterates until a valid stego image is generated. The number of iterations in the generation-validation process is typically less than 3 as our gradually strengthening prompt extending strategy gradually strengthens the prompt. The final iteration of adding relevant text is a powerful feature representation method. In practical experiments using data in Sec. 2.1, the number of iterations N satisfies $N \leq 3$ in most cases. 51% of stego image generation only requires one iteration. These images are mainly about characters which are common in the communities. 26.5% of stego image generation requires two iterations, mainly including images about regular scenes and food. By extending the prompt to highlight features that do not match, accurate extraction can be achieved. 19.5% of stego image generation requires three iterations, where the category mainly involves unusual scenes, artistic styles, products, and animals. In these images, some features may be too difficult to be accurately represented. Therefore the corresponding text added to the image is needed. For the remaining 3% of the images, more than three iterations are required for extremely indistinct feature options. Thus, N can be considered a small constant on average, *i.e.*, $O(N) = O(1)$. By combining the above components, the total computational complexity can be expressed as $O_{total} = O(C_{MLLM}) + O(m+n) + O(N) \approx O(C_{MLLM} + m + n)$. In our experiments, the time is 168.3575 seconds per image.

5. Diversity Requirements and Achievement

In Section 4.4 of the paper, we discuss the diversity and controllability using the same key and secret message. This configuration primarily addresses the need for flexible control over transmitted content based on the communication context during actual use. Since multiple stego images may be required in a single conversation, this section further focuses on short single conversation scenarios, *i.e.*, both the key and theme remain unchanged. In this situation, there may be two practical security requirements: (1) **Style Consistency Requirement**: In some scenarios, stego images may be considered as originating from the same source. If differences between images are too significant, it may arouse suspicion. Therefore, under this threat model, security demands the maintenance of a consistent visual style across multiple images. This ensures statistical and perceptual consistency. (2) **Diversity Requirement**: In another scenario, attackers may infer steganographic patterns by analyzing the similarity between stego images.

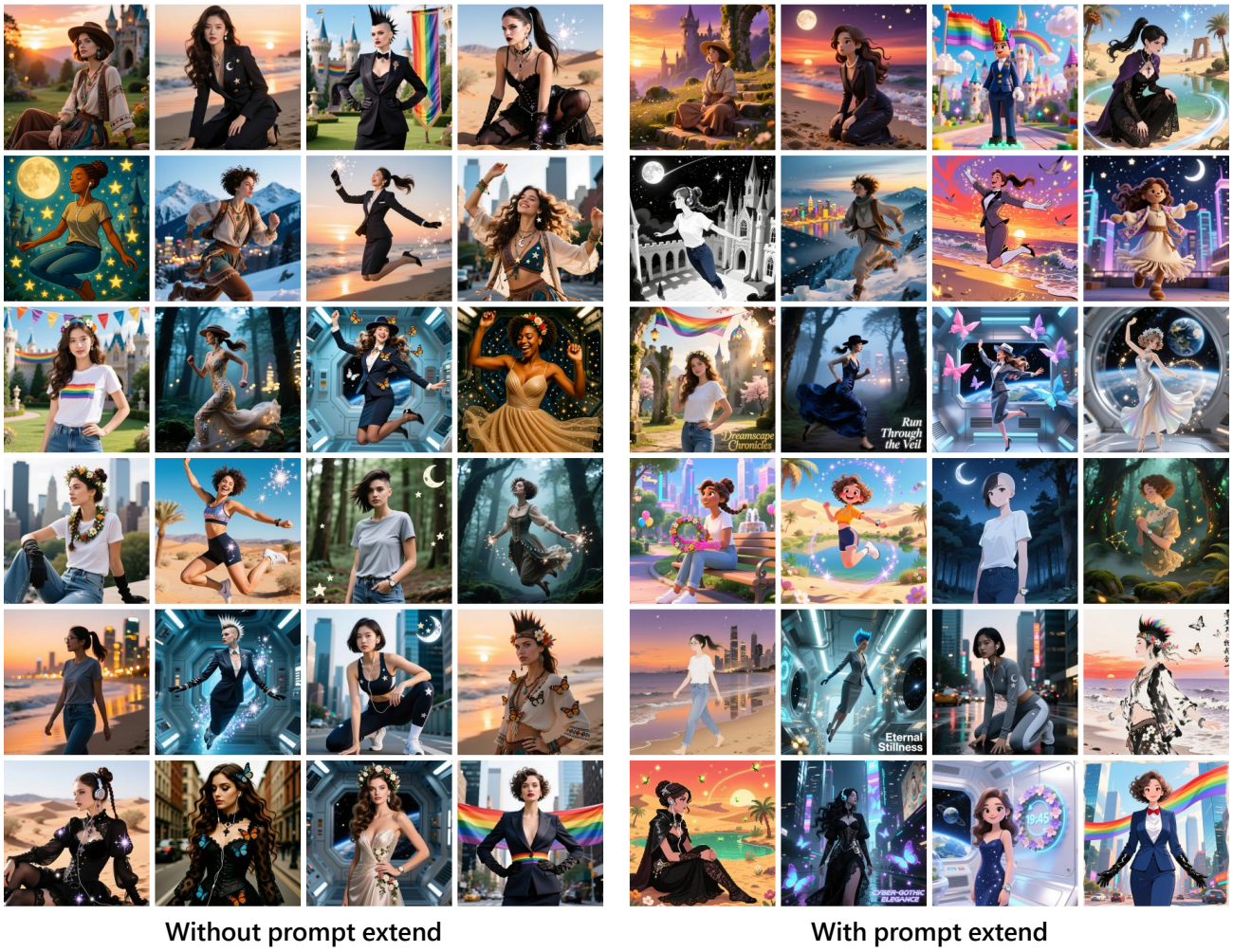

Figure 4. Experimental results with the same key and theme. In the "Without prompt extend" section and "With prompt extend" section, different images hide different secret messages in each section. For each corresponding position between the "Without prompt extend" section and the "With prompt extend" section, the same secret message is hidden.

For example, when the same secret message is hidden multiple times or different messages are hidden in the same theme, if the generated images are too similar, attackers can identify steganographic features through statistical analysis. In this case, security requires images to have differences in style, composition, details, and other aspects to reduce the risk of being inferred.

Our scheme can effectively adapt to both requirements. As shown in Fig. 4, we generate the same mapping table using the same key and theme, and then create stego images based on this mapping table. To meet the style consistency requirement, we can achieve stylistic consistency by standardizing the format of the prompt for image generation as shown in the "Without prompt extend" section of Fig. 4. In our scheme, this can be achieved by using the original image feature combination directly without prompt extending. Furthermore, we can leverage large language models to refine the original image feature combination or add style prompts to satisfy the diversity requirement. As shown in the "With prompt extend" section of Fig. 4, different styles, compositions, and details can be achieved. Even for images that are theoretically highly similar, such as those with the same key, theme, and secret messages, differences can be achieved.

6. Robustness comparison

In Section 4.7 of the paper, we conducted tests under various image attack conditions as shown in Table 5 of the paper. The actual effects of these image attacks can be seen in Fig. 5. To ensure fair comparison, we have also summarized the extraction accuracy of representative schemes under the attacks given in the original paper. Since some schemes used different datasets in their respective implementations, we averaged the results of the different datasets for each scheme to obtain the overall robustness performance. The comparison results are shown in Tab. 3. It should be specifically noted that in SCTTI [6], the secret messages are directly mapped to the PNG info metadata. SCTTI [6] assumes that PNG info is

Table 3. Robustness comparison of MCIS schemes. The data of previous work are directly extracted from the original papers. The values in the table represent the extraction accuracy.

Attack	parameters	MOR [8]	CI-CIS [7]	ETEHG [9]	NCIDC [15]	APAS [4]	GSIM [14]	SCTTI [6]	Ours
clean	None	100	100	100	100	100	100	100	100
JPEG compression	Quality factor = 10%	–	96.5	95.6	98.45	98.9	100	–	98.19
	Quality factor = 90%	86.6	–	97.3	100	100	100	–	99.56
Gaussian Noise	Mean = 0, variance = 0.001	68.4	65.5	95.3	99.6	99.3	100	–	98.94
Salt and pepper noise	Density = 0.001	84.7	67	95.6	99.8	98.8	100	–	99.17
Speckle noise	Mean = 0, variance = 0.01	82.9	65.5	95.1	98.8	99.3	100	–	98.86
Gaussian filtering	Window size = 3×3	86.8	96.5	91.2	100	98.2	100	–	99.64
Mean filtering	Window size = 3×3	82.1	96.5	90.1	99.2	98.4	100	–	99.33
Median filtering	Window size = 3×3	84	99	91.1	100	98.2	100	–	99.33
Color histogram equalization	None	76.2	97	91.2	92.95	85.2	98.23	–	98.47
Gamma correction	Factor = 0.8	–	97	96.3	100	98.7	100	–	99.08
Center cropping	Ratio = 10%	41	–	85.7	–	98.4	100	–	98.67
Edge cropping	Ratio = 10%	84.6	–	98.8	97.25	99.4	99.83	–	98.89
Size invariant rotation	Angle = 10°	–	–	89	–	99.4	100	–	98.31
Content invariant rotation	Angle = 10°	77.4	96	–	98.45	–	99.54	–	98.22
Translation	Distance = [40, 25]	81.1	–	88.7	–	83.3	99.97	–	98.61
Scaling	Scale = 0.5	–	–	–	100	–	100	-	98.61
	Scale = 1.5	–	–	–	100	–	100	-	99.56
Shear transformation	Angle = $[10^\circ, 10^\circ]$	–	–	–	–	–	–	–	98.00
Affine transformation	Rotation angle = 10°, scale = 1.5, shear angle = $[5^\circ, 5^\circ]$, translation distance = [40, 25]	–	–	–	–	–	–	–	97.75
Perspective transformation	Point offset = $\begin{bmatrix} [200, 0] & [-200, 0] \\ [0, 0] & [0, 0] \end{bmatrix}$	–	–	–	–	–	–	–	98.58
Screen-shooting	Distance = 35 cm	–	–	–	–	–	–	–	94.31

robust; however, this claim is incorrect. PNG info is only preserved when the output images remain completely unmodified. Any attack on the image will destroy the metadata. As shown in Tab. 3, most MCIS schemes maintain good extraction accuracy under attacks. Our scheme achieves performance comparable to other advanced schemes under diverse attacks. Even under attacks not discussed in previous works, *i.e.*, shearing, affine transformation, perspective transformation, and screen-shooting, our scheme maintains high usability.

7. Discussion and Future work

Our scheme achieves competitive results and demonstrates significant practical potential. However, several work can be further investigated, reflecting both the challenges and opportunities in coverless steganography with MLLMs assistance.

Improving MLLMs universality and determinism. Current implementation has certain requirements in terms of universality across different MLLMs, as consistent mapping requires deterministic and stable model behavior. Although achieving complete universality across all MLLMs is beyond the scope of the paper, it would be an exciting advancement if steganography could be realized only through user access to MLLMs and image generators on web pages. Our scheme lays the foundation for this vision by enabling universal black-box integration with image generators and partial black-box access to MLLMs. To achieve this goal, ensuring the determinism of MLLM outputs is a must. However, it remains an open challenge at present [5]. In our scheme, we have provided several user-level strategies to enhance determinism. Further investigation into the causes of non-determinism and robust deterministic mechanisms [5] would be valuable.

Enhancing richness of MFOSs. Another valuable direction for future work lies in analyzing and enhancing the semantic richness of MFOSs. Both the number of distinguishable options within a single dimension and the number of distinguishable dimensions within a single theme are naturally limited. The bounded output capability of MLLMs may limit the richness of MFOSs to some extent as well. Our experiments verified the feasibility and competitive security of the proposed scheme and showed no significant similarity among MFOSs after standardization. However, further research on MFOS richness remain meaningful. One potential approach is to incorporate real-world knowledge as additional few-shot prompts. Nevertheless, this may increase system complexity, and ensuring that these added prompts are effectively recognized without inadvertently constraining diversity is a challenge.

Expanding information capacity. Our scheme has achieved the best information capacity compared to prior schemes, and is suitable for secure transmission of data such as encryption keys, authentication codes, and identity information. To further enhance capacity, multi-channel steganography presents a promising direction, but it may compromise security by introducing new ways of information leakage. A systematic exploration of this trade-off between capacity and security would provide valuable insights in future designs. Extending our scheme to other modalities to expand information capacity is appealing as well. With the rapid advancement of text-to-video and text-to-3D generation, our scheme has evident potential in these domains.

Figure 5. Image attacks.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu,

Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 1

[2] Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5): 1181–1193, 2019. 3

[3] Xiaoqing Deng, Bolin Chen, Weiqi Luo, and Da Luo. Fast and effective global covariance pooling network for image steganalysis. *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019. 3

[4] Bobiao Guo, Ping Ping, and Feng Xu. Highly robust and diverse coverless image steganography against passive and active steganalysis. *IEEE Transactions on Dependable and Secure Computing*, pages 2771–2787, 2025. 4, 5, 7

[5] Horace He and Thinking Machines Lab. Defeating nondeterminism in llm inference. *Thinking Machines Lab: Connectionism*, 2025. 1, 7

[6] Xiao Li, Liquan Chen, Tong Fu, Zhangjie Fu, and Yuan Gao. Coverless image steganography based on semantic-controlled text-to-image generation. *IEEE TCSVT*, page 1, 2025. 3, 4, 5, 6, 7

[7] Qiang Liu, Xuyu Xiang, Jiaohua Qin, Yun Tan, and Qin Zhang. A robust coverless steganography scheme using camouflage image. *IEEE TCSVT*, pages 4038–4051, 2022. 4, 5, 7

[8] Yuanjing Luo, Jiaohua Qin, Xuyu Xiang, and Yun Tan. Coverless image steganography based on multi-object recognition. *IEEE TCSVT*, pages 2779–2791, 2020. 3, 4, 5, 7

[9] Laijin Meng, Xinghao Jiang, Zhenzhen Zhang, Zhaohong Li, and Tanfeng Sun. A robust coverless image steganography based on an end-to-end hash generation model. *IEEE TCSVT*, pages 3542–3558, 2023. 4, 5, 7

[10] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708–712, 2016. 3

[11] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. 2

[12] Jian Ye, Jiangqun Ni, and Yang Yi. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):2545–2557, 2017. 3

[13] Weike You, Hong Zhang, and Xianfeng Zhao. A siamese cnn for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 16:291–306, 2021. 3

[14] Qinghua Zhang and Fangjun Huang. Robust generative steganography based on image mapping. *IEEE TCSVT*, page 1, 2024. 3, 4, 5, 7

[15] Liming Zou, Jing Li, Wenbo Wan, Q. M. Jonathan Wu, and Jiande Sun. Robust coverless image steganography based on neglected coverless image dataset construction. *IEEE TMM*, pages 5552–5564, 2022. 4, 5, 7