Supplementary Material for Effective Message Hiding with Order Preserving Mechanisms

This appendix is organized as follows.

A. Details of the kernel size experiment conducted using SteganoGAN [36] and ChatGAN [25].

B. Experiments investigating the selection of keys, values, and queries for GMIF.

C. The detailed configuration of StegaFormer.

D. Detailed settings for the quantitative experiment.

E. Additional qualitative comparisons between StegaFormer and previous approaches [25, 36]. Qualitative examples for our models from 1 BPP to 8 BPP.

F. Detailed settings for steganalysis experiments.

A Kernel Size Experiment

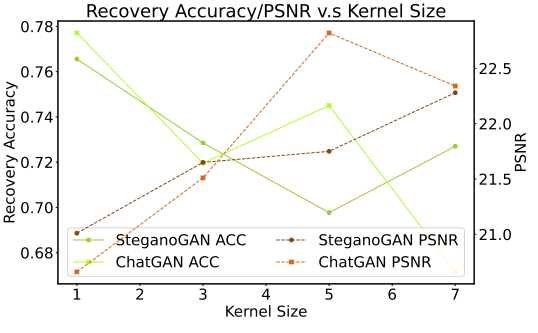

Figure 7: The effects of different kernel size to the recovery accuracy and imperceptibility.

As depicted in Fig. 7, we conduct several experiments related to the kernel size of the CNN-based approach to investigate its limitations. The experiments for SteganoGAN [36] is based on the officially released code. We choose the base model of SteganoGAN as it avoids the complexities of other variants that involve embedding messages at multiple convolutional stages, making it difficult to discern how message features are encoded into the image features. Since there is no official code released for ChatGAN [25], our experiments for this approach is based on our own implementations. We integrate the channel attention module into the SteganoGAN as described in the paper of ChatGAN. All models are trained using the DIV2K dataset with 4 BPP message payload for 20 epochs, with all other configurations remaining unchanged, except for the kernel size of these two models.

B Selection of Query, Key and Value for GMIF

While it is possible to achieve the message hiding task by using the default configuration of W-MSA, the fusion-related researches in other domain[3, 15, 16] using attention mechanism do not agree on the source of features for the query, key and value. To clarify this question, we conduct several experiments using COCO dataset at 4 BPP message capacity to validate different combinations of query, key, and value, searching for the best combination to achieve higher performance. As shown in Tab. 5, the GMIF, when using message features as a query, fails to conceal the message within the cover image, leading to a mere 50% message accuracy. Despite achieving high PSNR and SSIM values, the model prioritizes reconstructing an exact replica of the cover image over performing the message hiding task. The underlined row in Tab. 5 indicates the final configuration adopted by our paper, *i.e.*, using features from the image as a query and sum of image and message feature as key and value. This combination achieves the best message recovery accuracy and imperceptibility.

Query	Key and Value	ACC	PSNR	SSIM
MSG	MSG+IM	50.00%	168.8	1.0
MSG	IM	99.23%	41.35	0.9863
IM	MSG+IM	99.27%	41.87	0.9877
IM	MSG	99.22%	41.43	0.9863
MSG+IM	MSG+IM	99.07%	41.40	0.9862

Table 5: The comparison between different configurations of GMIF. MSG and IM represent the message feature and image feature respectively. MSG+IM denotes the summation of image and message features. The underline indicates the configuration adopted in our paper.

C Detailed Configurations of StegaFormer

The message concealment pipeline comprises three OPME modules for order-preserving message encoding. Each OPME consists of one MHSA, with 2 heads. Additionally, there are three GMIF modules to fuse the message-image features. Each GMIF includes one W-MSA, with the number of heads set to 2 and the window size set to 16. Regarding the message recovery pipeline, there are four W-MSA modules, all configured to 16 window size and 2 attention heads. Furthermore, one OPMD serving as the message head for recovering the secret message in the message recovery process. The number of heads is set to 2 for MHSA inside OPMD. The detailed configuration of the message recovery pipeline is listed in Tab. 6.

The number of parameters in StegaFormer is solely determined by the length of message segment L_{ms}. As mentioned in the main paper, the typical settings for L_{ms} are 16, 32, 48, and 64. On the other hand, increasing the range of message element N_{r} does not affect the number of parameters in StegaFormer. The typical number of parameters for our StegaFormer at different values of L_{ms} are listed in Tab. 7. We also list the Flops and number of parameters at typical 1 BPP in Tab. 8.

There are two approaches to increase the message hiding capacity in our method: increasing the length of message segments L_{ms} and expanding the range of message elements N_r. When $N_r = 1$, typical choices for L_{ms} include 16, 32, 48, and 64, corresponding to message capacities of 1, 2, 3, and 4 BPP, respectively. N_r can also be increased to achieve

higher message capacities. For example, $N_r = 2^i - 1, \quad$ where $i = 2, 3, 4$, represents 2, 3, and 4 bits per message element, respectively. Combining these two factors can lead to a higher message capacity for our models.

We list the experiment results related to different configurations of L_{ms} and N_{r} we have tested in searching for maximum message capacity in Tab. 9 and underlined the listed combinations in the main paper. All the experiments are trained with COCO dataset.

Module Name	Sub Module	Number of Layers	Number of Heads	Window Size	Output Feature Dimension
W-MSA 1	NA	1	2	16	$2 \times L_{\text{ms}}$
W-MSA 2	NA	1	2	16	$4 \times L_{\text{ms}}$
W-MSA 3	NA	1	2	16	$8 \times L_{\text{ms}}$
W-MSA 4	NA	1	2	16	$8 \times L_{\text{ms}}$
OPMD	MHSA	1	2	NA	$8 \times L_{\text{ms}}$
	MLP	1	NA	NA	L_{ms}

Table 6: Detailed configurations for message recovery pipeline. The number added to W-MSA represents the order of W-MSA in the message recovery process from right to left. NA denotes not available.

(L_{ms}, N_r)	Capacity	Encoder	Decoder	(L_{ms}, N_r)	Capacity	Encoder	Decoder	(L_{ms}, N_r)	Capacity	Encoder	Decoder
(16, 1)	1 BPP	10.75M	2.39M	(48, 1)	3 BPP	48.25M	10.70M	(48, 3)	6 BPP	48.25M	10.70M
(32, 1)	2 BPP	21.46M	4.76M	(64, 1)	4 BPP	85.75M	19.00M	(32, 15)	8 BPP	21.46M	4.76M

Table 7: Number of parameters of StegaFormer from 1 to 8 BPP message capacity. M denotes a million parameters.

Method	FLOPs (G)	# Param (M)
StegaFormer	74.3	13.14
Message Concealment of StegaFormer	58.7	10.75
Message Recovery of StegaFormer	15.6	2.39

Table 8: Flops and number of parameters of StegaFormer in 1 BPP message capacity.

Configuration list					Configuration list				
$(L_{\text{ms}}, N_{\text{r}})$	BPP	ACC	PSNR	SSIM	$(L_{\text{ms}}, N_{\text{r}})$	BPP	ACC	PSNR	SSIM
(16,1)	1	99.95%	47.83	0.9969	(16,7)	3	98.72%	42.86	0.9890
(32,1)	2	99.85%	45.30	0.9943	(32,7)	6	95.03%	36.72	0.9673
(48,1)	3	99.68%	43.37	0.9914	(48,7)	9	78.45%	35.50	0.9432
(64,1)	4	99.27%	41.87	0.9877	(16,15)	4	96.32%	40.33	0.9799
(16,3)	2	99.61%	44.98	0.9934	(32,15)	8	91.78%	34.70	0.9508
(32,3)	4	97.46%	41.37	0.9847	(48,15)	12	66.31%	30.37	0.8673
(48,3)	6	95.65%	40.37	0.9803					

Table 9: The experiment results of StegaFormer using different configurations of L_{ms} and N_{r}. The selected configurations are underlined.

D Detailed Configurations of Experiments

The training process of our model consists of 100,000 iterations, with a batch size of 2. We set $\lambda_1 = 1 \times 10^{-4}$ and $\lambda_2 = 1 \times 10^{-6}$ to balance different losses. For comparison, we set the image size to 256×256 for all models. All other settings remain as in the officially released models. We utilize the official models of SteganoGAN and LISO [9] since they are publicly

available. We use our own inplamentation of ChatGAN based on SteganoGAN since there is no officially released code.

E More Qualitative Examples

More qualitative comparisons of residual and stego image generated by StegaFormer and other two methods are in Fig. 8. Additionally, more comparisons of residual and stego images generated by StegaFormer at different message capacities are in Fig. 9.

F Experiments of Steganalysis

As described in Sec. 4.3. We conduct steganalysis experiments using SiaStegNet [34]. Among all the experiments, we use COCO dataset for this experiments. We generate 25,000 training sets consisting of cover-stego image pairs and 500 testing pairs using listed approaches and our model. SiaStegNet is trained for 2 epochs. We do not continue training SiaStegNet for more epochs since the Siamese approach is highly effective when cover and stego image pairs are available for training. Training for more than 2 epochs would result in a 100% detection rate for all the methods. The aim of this experiment is to demonstrate the superiority of security of our approach.