LD-RoViS: Training-free Robust Video Steganography for Deterministic Latent Diffusion Model

Xiangkun Wang[1,2] **Kejiang Chen**[1,2]* **Lincong Li**[1,2] **Weiming Zhang**[1,2] **Nenghai Yu**[1,2]

[1]University of Science and Technology of China, China
[2]Anhui Province Key Laboratory of Digital Security, China

wangxiangkun@mail.ustc.edu.cn chenkj@ustc.edu.cn

Abstract

Existing video steganography methods primarily embed secret information by modifying video content in the spatial or compressed domains. However, such methods are prone to distortion drift and are easily detected by steganalysis. Generative steganography, which avoids direct modification of the cover data, offers a promising alternative. Despite recent advances, most generative steganography studies focus on images and are difficult to extend to videos because of compression-induced distortions and the unique architecture of video generation models. To address these challenges, we propose LD-RoViS, a training-free and robust video steganography framework for the deterministic latent diffusion model. By modulating implicit conditional parameters during the diffusion process, LD-RoViS constructs a dedicated steganographic channel. Additionally, we introduce a novel multi-mask mechanism to mitigate errors caused by video compression and post-processing. The experimental results demonstrate that LD-RoViS can embed approximately 12,000 bits of data into a 5-second video with an extraction accuracy exceeding 99%. Our implementation is available at `https://github.com/xiangkun1999/LD-RoViS`.

1 Introduction

In the contemporary digital era, the rapid spread and extensive sharing of information have brought increasing attention to information security. Steganography [1], as a key technique for information hiding, aims to embed secret data within normal data to enable covert communication. Among various media, video has become a particularly valuable cover for steganography because of its high complexity and large capacity for hiding data [2]. In contrast, steganalysis [3], as an adversarial technique of steganography, seeks to detect the presence of hidden information.

Early video steganography methods can be broadly categorized into spatial domain methods and transform domain methods. Spatial domain methods [4, 5, 6, 7, 8, 9] embed information by directly modifying pixel values in video frames. Transform domain methods [10, 11, 12, 13, 14, 15, 16], on the other hand, leverage transformations such as DWT and DCT to embed data in the frequency domain, enhancing robustness against video encoding. Given the advantages of compressed video in storage and transmission, researchers have also explored compressed domain steganography [17, 18, 19, 20, 21, 22, 23], embedding messages into elements such as DCT coefficients, motion vectors, and prediction modes during video compression. While effective in certain scenarios, these traditional methods inevitably modify video content or encode content, making them vulnerable to distortion drift and steganalysis attacks [24, 25, 26].

With the rapid development of generative AI, many high-performance models has emerged, capable of generating realistic text, images, and video. Among them, video generation models based on

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

diffusion models have become a dominant paradigm, exemplified by advanced systems such as Sora [27], Gen-4 [28], Veo-2, HunyuanVideo [29], and Wan2.1 [30]. Simultaneously, AI-generated videos have become widespread on social media platforms, reshaping the data environment for video steganography. According to statistics from Zebracat AI[2], AI-generated videos account for 40% of video content on major social platforms. Recent studies have proposed generative steganography methods [31, 32, 33, 34, 35, 36, 37, 38, 39], which avoid directly modifying the cover data, and instead embed secret messages implicitly during the data generation process. These methods have shown strong resistance to steganalysis. However, current research in this area has focused almost exclusively on images, leaving generative video steganography largely unexplored.

Given that videos are composed of sequences of images, and that generative image steganography has matured considerably, an intuitive question arises: Can these methods be extended to videos? Some researchers have made early attempts in this direction. For example, [40] extends [31] into latent space and provides a simple implementation of video steganography based on Stable Video Diffusion [41]. Similarly, [42] employs a face-swapping model to embed messages into facial features, achieving robust video steganography. However, these methods are not compatible with most state-of-the-art (SOTA) video generation models, which face three main challenges: First, SOTA video generation models often rely on deterministic sampling to accelerate inference, making methods such as [31, 32, 33, 40]—which depend on random noise sampling—inapplicable. Second, most diffusion samplers are designed for one-directional generation and cannot achieve perfect invertibility, whereas [36, 37, 38, 39] map messages to the initial noise and depend on accurate inversion of the sampling process for extraction, making them unsuitable for non-reversible video diffusion models. Third, compression processing [43] imposed by video-sharing platforms presents additional challenges to the robustness of steganographic embedding.

To address these limitations, we propose LD-RoViS (**L**atent **D**iffusion-based **Ro**bust **Vi**deo **S**teganography), a training-free and robust steganography framework designed for deterministic latent diffusion models. By modulating the conditional parameters at the final time step of the diffusion process, we construct a steganographic channel to embed secret information into latent variables. Furthermore, to ensure robustness against video compression on social platforms, we introduce a multi-mask mechanism to identify and utilize the robust regions in the latent space for message embedding.

Our main contributions can be summarized as follows:

- We introduce LD-RoViS, the first training-free video steganography method for deterministic latent diffusion models.
- We design an implicit parameter modulation strategy to seamlessly integrate message embedding into the generation process.
- We propose a multi-mask mechanism to identify robust regions in latent space, enabling adaptive message embedding and enhancing resistance to compression-induced errors.
- We conduct extensive experiments to evaluate LD-RoViS, demonstrating its superior performance over existing methods in terms of capacity, robustness, and security.

2 Related Work

2.1 Diffusion Models

The core framework of diffusion models consists of two phases: forward diffusion and reverse denoising. The forward process incrementally corrupts data with Gaussian noise until it becomes pure noise, whereas the reverse process learns to predict and remove noise via neural networks to reconstruct the original data. Diffusion models have emerged as the dominant paradigm in generative models because of their high-quality generation capabilities. Current diffusion models can be categorized on the basis of their sampling strategies, as outlined below.

DDPM (Denoising Diffusion Probabilistic Model). The forward process of DDPM [44] follows a Markov chain for noise addition, whereas the reverse process trains a U-Net via variational inference.

[2] `https://www.zebracat.ai/post/ai-video-creation-statistics`

From one time step t to the previous time step $t-1$, the reverse sampling in DDPM can be expressed as:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \epsilon, \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{1}$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ is the noise predicted by the neural network, α_t and $\bar{\alpha}_t$ are predefined noise scheduler parameters, and σ_t is the standard deviation of the sampled Gaussian noise. Despite excellent generation quality, DDPM requires hundreds of sampling iterations, leading to low efficiency. Additionally, its reliance on stochastic sampling of ϵ makes it a non-deterministic model.

DDIM (Denoising Diffusion Implicit Model). To address DDPM's inefficiency, DDIM [45] employs a non-Markovian process and reparameterization to enable sampling with arbitrary step sizes. From a time step t to a time step s (where $s < t$), the DDIM sampling can be expressed as:

$$\mathbf{x}_s = \sqrt{\bar{\alpha}_s} \mathbf{f}_\theta(\mathbf{x}_t, t) + \sqrt{1-\bar{\alpha}_s - \sigma_s^2} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + \sigma_s \epsilon, \ \mathbf{f}_\theta(\mathbf{x}_t, t) = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}, \tag{2}$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ is the noise predicted by the neural network, $\bar{\alpha}_t$ and $\bar{\alpha}_s$ are the predefined noise scheduler parameters. When $\sigma_s \neq 0$, DDIM becomes non-deterministic; otherwise, DDIM is deterministic.

Flow Matching. Instead of relying on Markov chains, Flow Matching [46] formulates diffusion as the continuous evolution of vector fields. From a time step t to a time step $t - \Delta t$, the flow-based sampling in Flow Matching can be expressed as:

$$\mathbf{x}_{t-\Delta t} = \mathbf{x}_t - \Delta t \cdot \mathbf{f}_\theta(\mathbf{x}_t, t), \tag{3}$$

where $\mathbf{f}_\theta(\mathbf{x}_t, t)$ is an output of a neural network. Since Flow Matching does not involve the sampling of random variables, it is deterministic.

2.2 Diffusion-Based Generative Steganography

Steganographic methods based on diffusion models embed hidden information into the data synthesis process. Previous work mainly focused on image steganography, which supports image-level [34, 35] or message-level [31, 40, 33, 32] hiding. In this paper, we focus on message-level hiding.

StegaDDPM [32] performs steganographic embedding at the final time step of the reverse diffusion process. By partitioning the probability density function of the Gaussian distribution into equal-probability intervals, it constructs a reversible mapping from binary messages to Gaussian noise variables ϵ, thereby enabling message embedding within the generated images. However, this method is highly sensitive to lossy image transformations.

LDStega [33] extends StegaDDPM from the pixel space to the latent space, making it compatible with mainstream latent diffusion models. By mapping messages to a biased distribution, LDStega reduces the probability of extraction errors. In addition, it enhances robustness by performing a pre-encoding and decoding scheme on the latent variables.

Pulsar [31] adopts a different mapping strategy. At the final time step of the diffusion process, two noise samples, ϵ_1 and ϵ_2, are generated via two distinct secret keys. These are then mixed according to a binary message to produce the final noise added to the image. The receiver extracts the hidden message by comparing the distances between the stego image and the two reference images generated via the secret keys.

PSyDUCK [40] extends Pulsar from the pixel space to the latent space and investigates the impact of using two distinct keys at different diffusion timesteps. Moreover, it explores the applicability of the method in video generation models.

These methods heavily rely on non-deterministic sampling models. However, mainstream video generation models typically adopt deterministic sampling to enable faster inference and better controllability [30], rendering the above methods inapplicable. Moreover, the compression [43] applied by video-sharing platforms poses additional challenges to the robustness of video steganography methods.

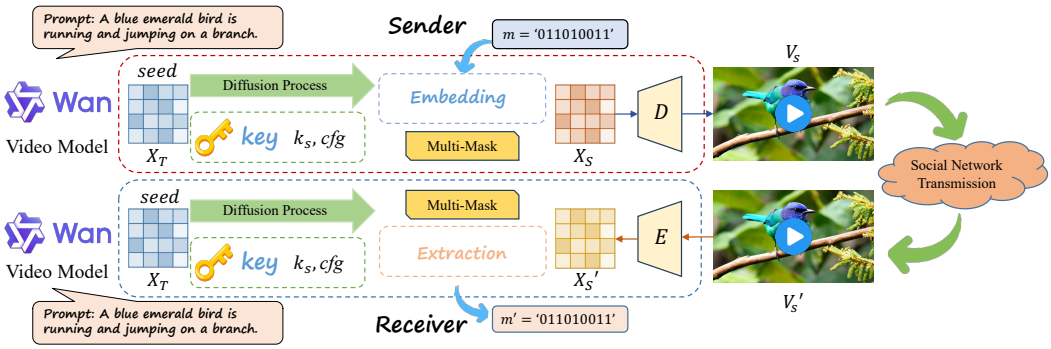

Figure 1: Overview of our proposed LD-RoViS.

3 Method

Our robust video steganography method is built upon a deterministic latent diffusion video model, enabling reliable message embedding and extraction even under challenging transmission conditions. The following sections provide a systematic overview of the proposed method.

3.1 Overview of LD-RoViS

As shown in Fig. 1, the sender and receiver share the same parameters: $prompt$, $seed$, k_s, and cfg, where $prompt$ refers to the text input fed into the video generation model to control the semantic content of the generated video. $seed$ is a random seed that controls the sampling of the initial noise X_T. The values k_s, and cfg denote different values of the classifier-free guidance (CFG) scale. The CFG scale controls how closely the generated content aligns with the input prompt.

On the sender's side, $prompt$, $seed$, and k_s are used to drive the diffusion process in the latent space up to the final time step. At this final time step, message embedding is performed via cfg and the message m, yielding the final latent variable X_s. This latent variable is then passed through the decoder D to generate the stego video V_s. The stego video is shared with the receiver via social platforms, and the received video is denoted as V_s'. The receiver encodes V_s' via the encoder E to obtain the latent variable X_s', and leverages the shared $prompt$, $seed$, and k_s to perform the same generation procedure as the sender. At the final time step, the receiver uses cfg and X_s' to extract the hidden message.

In the following, we use $V \in \mathbb{R}^{C\times F\times H\times W}$ to represent the video, and $X \in \mathbb{R}^{C'\times F'\times H'\times W'}$ to represent the latent variable, where C, F, H, W denote the number of channels, frames, height, and width of the video V, while C', F', H', W' represent the corresponding attributes in the latent space.

3.2 Message Embedding

The detailed framework of our proposed LD-RoViS is shown in Fig. 2, where "Shared" denotes the operations that need to be performed by both the sender and the receiver.

To embed a message, the sender first executes the "Shared" component. Specifically, the sender provides $prompt$ and $seed$ to the video generative model (denoted as G). The model G begins by sampling X_T from a Gaussian distribution using the given $seed$, and then performs $T-1$ denoising steps to obtain the latent representation X_1. We use $\textit{Diffuse}(\cdot)$ to represent the denoising process, then the above process can be formulated as:

$$X_1 = \textit{Diffuse}(G, X_T, k_s). \tag{4}$$

3.2.1 Parameter Modulation Strategy

The parameter modulation strategy lies at the core of our steganographic method. During each denoising step, the CFG scale controls the weighting between the predicted conditional and unconditional noise, formulated as:

$$\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) = pred_{uncond} + \text{CFG} \cdot (pred_{cond} - pred_{uncond}), \tag{5}$$

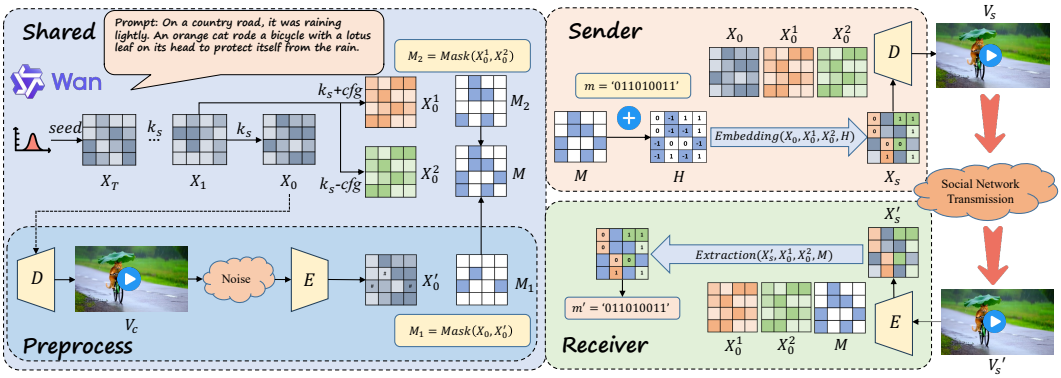

Figure 2: Framework of our proposed LD-RoViS.

where $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ denotes the noise predicted by G at each timestep, while $pred_{cond}$ and $pred_{uncond}$ represent the conditional and unconditional predicted noise, respectively.

Varying the CFG scale alters the trajectory of the diffusion process, resulting in different latent variables. This modulation strategy is equally applicable to deterministic diffusion models. On the basis of this insight, we can generate three different outputs—X_0, X_0^1, and X_0^2—from the same latent variable X_1 in the final denoising step by using different values of CFG k_s, $k_s + \text{cfg}$, and $k_s - \text{cfg}$, respectively. This can be expressed as:

$$X_i = \textit{Diffuse}(G, X_1, k_i), \quad \text{where } X_i \in \{X_0, X_0^1, X_0^2\},\ k_i \in \{k_s, k_s + \text{cfg}, k_s - \text{cfg}\}. \tag{6}$$

These latents exhibit subtle semantic differences while maintaining visual consistency with $prompt$, creating a discriminative latent pair for message encoding.

3.2.2 Multi-mask Mechanism

Before message embedding, we design a multi-mask mechanism to identify robust regions in the latent space, aiming to enhance the resilience of the steganographic method against operations such as video compression.

Invariance Mask M_1: This mask highlights latent positions that remain stable after encoder-decoder and noise addition. As shown in the "Preprocess" section of Fig. 2, we apply encoding, decoding, and noise to X_0 to simulate the distortion that may occur when the generated video is used in real-world scenarios. We use $\textit{Noise}(\cdot)$ to denote lossy processing such as video compression. The above process can be expressed as:

$$X_0' = E(\textit{Noise}(D(X_0))). \tag{7}$$

We then compute the L_1 distance between X_0 and X_0' as:

$$d_1(c, f, h, w) = \|X_0(c, f, h, w) - X_0'(c, f, h, w)\|, \tag{8}$$

where (c, f, h, w) indices valid positions within the spatial-temporal dimensions.

The invariance mask M_1 is defined as the set of position indices corresponding to the smallest $\tau_1\%$ values in d_1. We define this function as $\textit{Mask}(\cdot)$, formally expressed as:

$$M_1(c, f, h, w) = \textit{Mask}(d_1, \tau_1) = \begin{cases} 1 & \text{if } d_1(c, f, h, w) \in \text{top smallest } \tau_1, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

$M_1 = 1$ marks invariant regions resilient to codec distortions and noise.

Discriminative Mask M_2: This mask identifies latent positions where X_0^1 and X_0^2 differ the most, enabling reliable bit discrimination. Similarly, we compute L_1 distances between X_0^1 and X_0^2:

$$d_2(c, f, h, w) = \|X_0^1(c, f, h, w) - X_0^2(c, f, h, w)\|. \tag{10}$$

The discriminative mask M_2 is defined as the set of position indices corresponding to the largest $\tau_2\%$ values in d_2, formally expressed as:

$$M_2(c, f, h, w) = I - \textit{Mask}(d_2, 1 - \tau_2) = \begin{cases} 1 & \text{if } d_2(c, f, h, w) \in \text{top largest } \tau_2, \\ 0 & \text{otherwise,} \end{cases} \tag{11}$$

where I is the identity matrix.

Combined Mask M: The final mask is the dot product of M_1 and M_2, retaining positions that are both invariant ($M_1 = 1$) and discriminative ($M_2 = 1$):

$$M = M_1 \odot M_2, M \in \mathbb{R}^{C' \times F' \times H' \times W'}. \tag{12}$$

Positions where $M = 1$ (both masks are 1) are used for message embedding, ensuring robustness.

3.2.3 Embedding

On the basis of the analysis in Section 3.2.2, the steganographic capacity of LD-RoViS depends on τ_1 and τ_2. For a generated video V_s, assume that M_1 and M_2 are independent, the embedding capacity n is given by:

$$n = C' \times F' \times H' \times W' \times \tau_1 \times \tau_2, \tag{13}$$

where slight variations may occur due to ties in percentile rankings. Accordingly, we embed a binary message m of length n, and construct an embedding matrix H from m via the following rule:

$$H(c,f,h,w) = \textit{Transform}(M,m) = \begin{cases} -1 & \text{if } M(c,f,h,w) = 0, \quad \text{(non-embedding region)} \\ m_k & \text{if } M(c,f,h,w) = 1, \quad \text{(embedding region)} \end{cases} \tag{14}$$

where $m_k \in \{0,1\}$ is the k-th bit of m, which is filled row-wise from top-left to bottom-right.

The stego latent X_s is formed by mixing X_0, X_0^1, and X_0^2 according to H:

$$X_s(c,f,h,w) = Embedding(X_0, X_0^1, X_0^2, H) = \begin{cases} X_0(c,f,h,w) & \text{if } H(c,f,h,w) = -1, \\ X_0^1(c,f,h,w) & \text{if } H(c,f,h,w) = 0, \\ X_0^2(c,f,h,w) & \text{if } H(c,f,h,w) = 1. \end{cases} \tag{15}$$

Finally, X_s is decoded to the stego video $V_s = D(X_s)$ for transmission.

Algorithm 1 Message Embedding Algorithm

Input: $prompt$, $seed$, k_s, cfg, message m, thresholds τ_1, τ_2
Output: stego video V_s
Shared: G, D, E
  Sample X_T from $\mathcal{N}(0, \mathbf{I})$
  Perform T-step denoising:
  $X_0 \leftarrow \textit{Diffuse}(G, X_T, k_s)$
  $X_0^1 \leftarrow \textit{Diffuse}(G, X_1, k_s + \text{cfg})$
  $X_0^2 \leftarrow \textit{Diffuse}(G, X_1, k_s - \text{cfg})$
Preprocess:
  $X_0' \leftarrow E(\textit{Noise}(D(X_0)))$
  $M_1 \leftarrow \textit{Mask}(L_1(X_0, X_0'), \tau_1)$
  $M_2 \leftarrow I - \textit{Mask}(L_1(X_0^1, X_0^2), 1 - \tau_2)$
  $M \leftarrow M_1 \odot M_2$
Message Encoding:
  $H \leftarrow \textit{Transform}(M, m)$
Latent Mixing:
  For each (c,f,h,w) in latent space:
    if $H(c,f,h,w) = -1$ **then** $X_s(c,f,h,w) \leftarrow X_0(c,f,h,w)$
    elif $H(c,f,h,w) = 0$ **then** $X_s(c,f,h,w) \leftarrow X_0^1(c,f,h,w)$
    else $X_s(c,f,h,w) \leftarrow X_0^2(c,f,h,w)$
  $V_s \leftarrow D(X_s)$
return V_s

Algorithm 2 Message Extraction Algorithm

Input: $prompt$, $seed$, k_s, cfg, received video V_s', thresholds τ_1, τ_2
Output: recovered message m'
Shared: G, D, E
  Sample X_T from $\mathcal{N}(0, \mathbf{I})$
  Perform T-step denoising:
  $X_0 \leftarrow \textit{Diffuse}(G, X_T, k_s)$
  $X_0^1 \leftarrow \textit{Diffuse}(G, X_1, k_s + \text{cfg})$
  $X_0^2 \leftarrow \textit{Diffuse}(G, X_1, k_s - \text{cfg})$
Preprocess:
  $X_0' \leftarrow E(\textit{Noise}(D(X_0)))$
  $M_1 \leftarrow \textit{Mask}(L_1(X_0, X_0'), \tau_1)$
  $M_2 \leftarrow I - \textit{Mask}(L_1(X_0^1, X_0^2), 1 - \tau_2)$
  $M \leftarrow M_1 \odot M_2$
Stego Latent Extraction:
  $X_s' \leftarrow E(V_s')$
Message Decoding:
  Initialize empty message m'
  For each (c,f,h,w) where $M(c,f,h,w) = 1$:
    $d_1 \leftarrow \|X_s'(i,j) - X_0^1(i,j)\|$
    $d_2 \leftarrow \|X_s'(i,j) - X_0^2(i,j)\|$
    if $d_1 < d_2$ **then** append 0 to m'
    else append 1 to m'
return m'

3.3 Message Extraction

The receiver, equipped with the shared $prompt$ and parameters ($seed, k_s$, cfg), regenerates X_0, X_0^1, X_0^2, and M via the same steps ("Shared" section in Fig. 2). The received video V_s' is encoded to $X_s' = E(V_s')$.

For each position (c, f, h, w) where $M(c, f, h, w) = 1$, the receiver compares the L_1 distances between X_s' and X_0^1, X_0^2 to recover the message $m' = Extraction(X_s', X_0^1, X_0^2, M)$:

$$d_1'(c, f, h, w) = \|X_s'(c, f, h, w) - X_0^1(c, f, h, w)\|, \tag{16}$$

$$d_2'(c, f, h, w) = \|X_s'(c, f, h, w) - X_0^2(c, f, h, w)\|, \tag{17}$$

$$m_k' = \begin{cases} 0 & \text{if } d_1'(c, f, h, w) < d_2'(c, f, h, w) \; and \; M(c, f, h, w) = 1, \\ 1 & \text{if } d_1'(c, f, h, w) \geq d_2'(c, f, h, w) \; and \; M(c, f, h, w) = 1, \end{cases} \tag{18}$$

where $m_k' \in \{0, 1\}$ is the k-th bit of m', which is filled row-wise from top-left to bottom-right.

The embedding and extraction procedures are shown in Algorithm 1 and Algorithm 2, respectively. According to Equations 16, 17, and 18, the accuracy of message extraction is highly dependent on the difference between X_0^1 and X_0^2. To simplify the design while maintaining this distinguishability, we generate the two variables using symmetric CFG scales, *i.e.*, $k_s +$ cfg and $k_s -$ cfg, where cfg is a tunable hyperparameter controlling the modulation intensity.

4 Experiment

4.1 Experiment Setup

Our proposed LD-RoViS is built upon a deterministic latent video diffusion model. To this end, we adopt the T2V-1.3B model from Wan2.1 [30] as our video model. This model leverages Flow Matching sampling to efficiently produce high-quality videos. For evaluation, we use VidProM [47], a large-scale and diverse text-to-video prompt dataset. We randomly sample 100 prompts from VidProM and generate corresponding videos via Wan2.1. Each video has a resolution of 480×832, a duration of 5 seconds, and a frame rate of 16 fps, resulting in 81 frames per video. All subsequent experiments are conducted on these 100 prompts with $seed = 99, k_s = 5.0$ and run on four NVIDIA RTX A6000 GPUs, each with 48 GB of VRAM. In our experiments, we set the hyperparameters as $\tau_1 = 0.32$, $\tau_2 = 0.02$, and $cfg = 16$. Additional experiments and analysis of these hyperparameters can be found in the Appendix A, including experiments on a non-deterministic model (LTX-Video [48]).

4.2 Evaluation Metrics

We evaluate the performance of our steganographic method via the following metrics:

Accuracy (acc, %) is defined as the ratio of correctly recovered bits in m' to m. A higher acc indicates greater robustness.

Peak Signal-to-Noise Ratio (PSNR, dB) measures the perceptual similarity between two images; a higher PSNR indicates better image quality. In this work, the PSNR is computed as the average PSNR across all frames between V_c and V_s.

BRISQUE [49] is a no-reference image quality assessment metric commonly used to evaluate the perceptual quality of generated images. Lower BRISQUE scores correspond to higher perceptual quality. Here, we define it as the average BRISQUE score over all frames of V_s.

Capacity refers to the total number of data bits that can be embedded in the stego video V_s.

Error Rate (P_E, %) denotes the error rate in steganalysis detection. A higher P_E indicates better security of the steganographic method.

4.3 Quantitative Performance Evaluation

4.3.1 Metric Experiments

To evaluate the performance of the proposed method, we conducted experiments on 100 videos generated from diverse prompts. As shown in Table 1, we report four metrics: PSNR, BRISQUE

score, accuracy (acc), and embedding capacity. Each result is presented as the mean and standard deviation over the 100 test videos. We compare our method with three recent video steganography methods: two transform-domain methods (AQIM [12] and MEC_AQIM [13]), and a face-swapping-based generative method (RoGVSN [42]). For a fair comparison, we set the embedding capacity of AQIM and MEC_AQIM to 10,000 bits and tested them under the same conditions. Since RoGVSN only supports face-related video and does not allow for adjustable capacity, we selected 10 face-related videos from the test set for its evaluation. Additionally, the PSNR is not applicable to RoGVSN because of its face-swapping nature, and the corresponding values are left blank. The experimental results show that our method significantly outperforms the baselines in terms of visual quality, while maintaining high extraction accuracy and embedding capacity.

Table 1: Comparison of performance. The results are presented as the means and standard deviations.

Method	PSNR↑	BRISQUE↓	acc (%)↑	capacity↑
AQIM	34.81 ± 0.44	32.87 ± 6.06	$\mathbf{99.44 \pm 0.27}$	10000 (fixed)
MEC_AQIM	35.21 ± 0.47	32.71 ± 6.10	90.99 ± 5.90	10000 (fixed)
RoGVSN	–	49.53 ± 4.55	99.28 ± 0.38	729 (fixed)
Ours	$\mathbf{41.66 \pm 1.52}$	$\mathbf{28.90 \pm 6.05}$	99.17 ± 0.63	$\mathbf{11983 \pm 1446}$

4.3.2 Security Experiments

To evaluate the resistance of our proposed LD-RoViS against steganalysis attacks, we compare it with three baseline methods. Following the experimental setup in [12], we adopt two image steganalyzers, CovNet [24] and LWENet [25], and one video steganalysis feature extractor, SUPERB [50] combined with a linear classifier [51]. We test 100 generated cover-stego video pairs. For a fair comparison, the capacity of AQIM and MEC_AQIM is set to 10,000 bits. Since the capacity of RoGVSN is fixed, we follow its original design with a payload of 729 bits. Using ffmpeg, we decode these videos to obtain 8,100 pairs of cover and stego frames, with 4,000 used for training, 600 for validation, and 3,500 for testing. As shown in Table 2, LD-RoViS effectively resists steganalysis attacks, achieving error rates close to 50% (random guessing).

Table 2: P_E (%,↑) of steganalysis.

Method	SUPERB	CovNet	LWENet
AQIM	49.14	0.13	0.26
MEC_AQIM	47.32	0.01	1.07
RoGVSN	47.58	0.36	2.61
ours	**49.18**	**49.74**	**48.49**

Table 3: acc(%) under different compression and noise.

Method	-	CRF=18	CRF=23	CRF=27	noise	salt&pepper	brightness
AQIM	**99.44**	91.24	90.67	87.49	82.46	80.04	48.93
MEC_AQIM	90.99	82.83	82.29	78.87	72.83	71.60	50.31
RoGVSN	99.28	**97.42**	**97.06**	**97.04**	**96.20**	94.45	96.05
ours	99.17	95.89	93.70	91.67	92.82	**98.72**	**99.02**

4.3.3 Robustness Experiments

To evaluate the robustness of LD-RoViS against video compression and noise, we conducted the following experiments. The most common processing method on social media platforms is H.264 compression, where the Constant Rate Factor (CRF) is a key quality control parameter, typically set to 23. In our experiments, we tested both CRF=18 and CRF=27. Additionally, we applied common noise perturbations: "noise" refers to Gaussian noise with a standard deviation of 0.05, "salt&pepper[3]" denotes impulse noise with a probability of 0.01, and "brightness" represents an increase of 0.1 in the HSV color space. As shown in Table 3, LD-RoViS demonstrates strong robustness to brightness changes and salt-and-pepper noise, while maintaining over 90% extraction accuracy under other lossy conditions. Although RoGVSN is more robust against H.264 compression and Gaussian noise, it is important to note that its embedding capacity is only 729 bits, whereas LD-RoViS supports approximately 12,000 bits.

4.4 Subjective performance evaluation

To evaluate the impact of steganographic embedding on video visual quality, we present several visual examples in Fig. 3. Using FFmpeg, we decode both cover and stego videos and extract the middle

[3]Salt-and-pepper noise is applied in an image-level manner, where a single spatial impulse mask is shared across frames with an overall probability of 0.01.

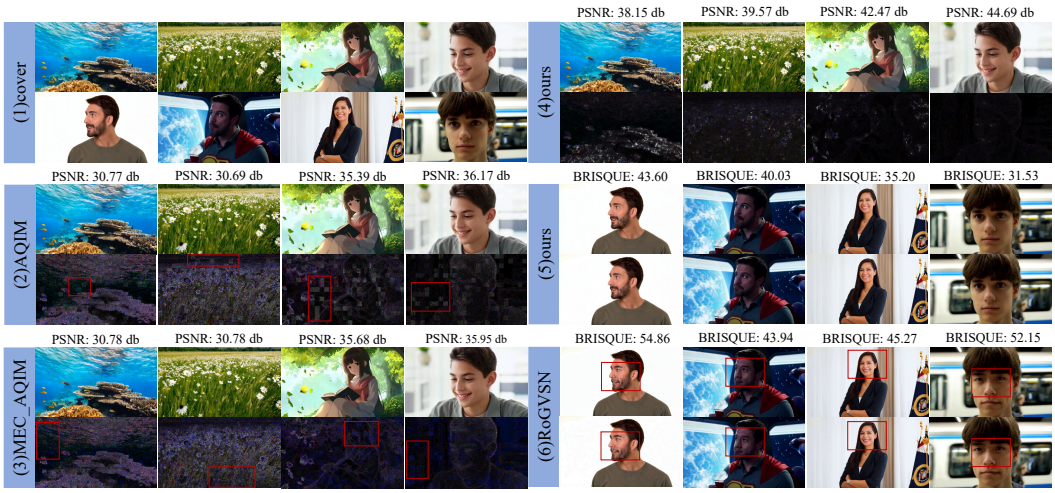

Figure 3: Visual comparisons between our method and baselines. In (2)–(4), the second row shows the pixel differences between the cover and stego frames (brightness increased by 10×), whereas in (5)–(6), the second row shows the temporally adjacent frames.

frame (the 41st frame). In SubFig. 3 (2), (3), and (4), the top row shows the stego frames, whereas the bottom row shows the pixel-wise differences between the stego and cover frames (brightness enhanced 10× for better visibility). As shown, traditional transform-domain methods are sensitive to video encoding and introduce noticeable block artifacts. In contrast, our method avoids direct modification of video content, resulting in only minor pixel differences in texture-rich regions. In SubFig. 3 (5) and (6), the top row shows the stego frame, whereas the bottom row displays the subsequent frame in temporal order (i.e., the 42nd frame). Since RoGVSN embeds messages via face-swapping, noticeable blur and distortion are observed in facial regions, leading to lower perceptual quality. These results demonstrate that our method balances effective steganography with minimal perceptual impact, outperforming baselines in preserving video quality for real-world applications.

4.5 Ablation Studies

To further investigate the effectiveness of the multi-mask mechanism, we conducted ablation studies on the two masks M_1 and M_2. Specifically, we evaluated three different variants of the full model, with their differences summarized in Table 4. Each variant was tested on 10 generated videos, and the average results are reported in Table 5. The experimental results show that removing either M_1 or M_2 leads to a significant decrease in accuracy, confirming the effectiveness of M_1 in identifying invariant regions and M_2 in identifying discriminative regions within the latent space. Their combined effect enables the identification of robust areas in the latent space to achieve robust steganography.

Table 4: Ablation variants.

Method	Mask M_1	Mask M_2
variant#1	×	×
variant#2	✓	×
variant#3	×	✓
ours	✓	✓

Table 5: Performance of different variants.

Method	acc(%)↑	PSNR(db)↑	BRISQUE↓	capacity(bits)↑
variant#1	62.67	35.39	30.55	**1935111**
variant#2	75.46	37.49	29.47	617913
variant#3	88.59	40.53	29.01	41132
ours	**99.17**	**41.66**	**28.90**	11983

5 Conclusion

In this work, we present a robust video steganography method for deterministic latent diffusion models. Our method innovatively constructs a steganographic channel by leveraging the classifier-free guidance (CFG) scale of diffusion models. In addition, we introduce a multi-mask mechanism based on adversarial encoding-decoding and video compression perturbations to identify invariant and distinguishable regions in the latent space, thereby ensuring robustness. To the best of our knowledge,

this is the first steganographic method for deterministic latent diffusion, which achieves significant advantages in both embedding capacity and security compared with existing video steganography methods.

6 Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant U2336206, Grant 62472398, Grant U2436601, and Grant 62402469.

References

[1] Ross J Anderson and Fabien AP Petitcolas. On the limits of steganography. *IEEE Journal on selected areas in communications*, 16(4):474–481, 1998.

[2] Jayakanth Kunhoth, Nandhini Subramanian, Somaya Al-Maadeed, and Ahmed Bouridane. Video steganography: recent advances and challenges. *Multimedia Tools and Applications*, 82(27):41943–41985, 2023.

[3] Jessica Fridrich and Miroslav Goljan. Practical steganalysis of digital images: state of the art. *security and Watermarking of Multimedia Contents IV*, 4675:1–13, 2002.

[4] Kousik Dasgupta, JK Mandal, and Paramartha Dutta. Hash based least significant bit technique for video steganography (hlsb). *International Journal of Security, Privacy and Trust Management (IJSPTM)*, 1(2):1–11, 2012.

[5] Vivek Kumar Jha, Srilekha Mukherjee, Subhajit Roy, and Goutam Sanyal. Video steganography technique using factorization and spiral lsb methods. In *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, pages 315–320. IEEE, 2017.

[6] Ramadhan J Mstafa, Younis Mohammed Younis, Haval Ismael Hussein, and Muhsin Atto. A new video steganography scheme based on shi-tomasi corner detector. *IEEE Access*, 8:161825–161837, 2020.

[7] Mritha Ramalingam and Nor Ashidi Mat Isa. A steganography approach over video images to improve security. *Indian Journal of Science and Technology*, 8(1):79, 2015.

[8] Hamdy M Kelash, Osama F Abdel Wahab, Osama A Elshakankiry, and Hala S El-sayed. Utilization of steganographic techniques in video sequences. *International Journal of Computing and Network Technology*, 2(01), 2014.

[9] Sneha Khupse and Nitin N Patil. An adaptive steganography technique for videos using steganoflage. In *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, pages 811–815. IEEE, 2014.

[10] Anush Kolakalur, Ioannis Kagalidis, and Branislav Vuksanovic. Wavelet based color video steganography. *International Journal of Engineering and Technology*, 8(3):165, 2016.

[11] Meenu Suresh and I Shatheesh Sam. Optimal wavelet transform using oppositional grey wolf optimization for video steganography. *Multimedia Tools and Applications*, 79:27023–27037, 2020.

[12] Pingan Fan, Hong Zhang, Yifan Cai, Pei Xie, and Xianfeng Zhao. A robust video steganographic method against social networking transcoding based on steganographic side channel. In *Proceedings of the 2020 ACM workshop on information hiding and multimedia security*, pages 127–137, 2020.

[13] Pingan Fan, Hong Zhang, and Xianfeng Zhao. Adaptive qim with minimum embedding cost for robust video steganography on social networks. *IEEE Transactions on Information Forensics and Security*, 17:3801–3815, 2022.

[14] Ramadhan J Mstafa and Khaled M Elleithy. An ecc/dct-based robust video steganography algorithm for secure data communication. *Journal of Cyber Security and Mobility*, pages 167–194, 2016.

[15] Meenu Suresh and I Shatheesh Sam. High secure video steganography based on shuffling of data on least significant dct coefficients. In *2018 Second international conference on intelligent computing and control systems (ICICCS)*, pages 877–882. IEEE, 2018.

[16] Mritha Ramalingam and Nor Ashidi Mat Isa. Video steganography based on integer haar wavelet transforms for secured data transfer. *Indian Journal of Science and Technology*, 7(7):897–904, 2014.

[17] Yun Cao, Yu Wang, Xianfeng Zhao, Meineng Zhu, and Zhoujun Xu. Cover block decoupling for content-adaptive h. 264 steganography. In *Proceedings of the 6th ACM workshop on information hiding and multimedia security*, pages 23–30, 2018.

[18] Yunxia Liu, Leiming Ju, Mingsheng Hu, Hongguo Zhao, Suimin Jia, and Zhijuan Jia. A new data hiding method for h. 264 based on secret sharing. *Neurocomputing*, 188:113–119, 2016.

[19] Si Liu, Yunxia Liu, Cong Feng, Hongguo Zhao, and Yu Huang. A hevc steganography method based on qdct coefficient. In *Intelligent Computing Methodologies: 16th International Conference, ICIC 2020, Bari, Italy, October 2–5, 2020, Proceedings, Part III 16*, pages 624–632. Springer, 2020.

[20] Dinh-Chien Nguyen, Thai-Son Nguyen, Fang-Rong Hsu, and Hsieh-Yung Hsien. A novel steganography scheme for video h. 264/avc without distortion drift. *Multimedia Tools and Applications*, 78:16033–16052, 2019.

[21] Yiming Xue, Jie Zhou, Hao Zeng, Ping Zhong, and Juan Wen. An adaptive steganographic scheme for h. 264/avc video with distortion optimization. *Signal Processing: Image Communication*, 76:22–30, 2019.

[22] Hong Zhang, Yun Cao, and Xianfeng Zhao. Motion vector-based video steganography with preserved local optimality. *Multimedia Tools and Applications*, 75:13503–13519, 2016.

[23] Lingyu Zhang and Xianfeng Zhao. An adaptive video steganography based on intra-prediction mode and cost assignment. In *International workshop on digital watermarking*, pages 518–532. Springer, 2016.

[24] Xiaoqing Deng, Bolin Chen, Weiqi Luo, and Da Luo. Fast and effective global covariance pooling network for image steganalysis. In *Proceedings of the ACM workshop on information hiding and multimedia security*, pages 230–234, 2019.

[25] Shaowei Weng, Mengfei Chen, Lifang Yu, and Shiyao Sun. Lightweight and effective deep image steganalysis network. *IEEE Signal Processing Letters*, 29:1888–1892, 2022.

[26] 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, 2018.

[27] 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. Video generation models as world simulators. 2024.

[28] Shakiba Kheradmand, Delio Vicini, George Kopanas, Dmitry Lagun, Kwang Moo Yi, Mark Matthews, and Andrea Tagliasacchi. Stochasticsplats: Stochastic rasterization for sorting-free 3d gaussian splatting. *arXiv preprint arXiv:2503.24366*, 2025.

[29] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[30] Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[31] Tushar M Jois, Gabrielle Beck, and Gabriel Kaptchuk. Pulsar: Secure steganography for diffusion models. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 4703–4717, 2024.

[32] Yinyin Peng, Donghui Hu, Yaofei Wang, Kejiang Chen, Gang Pei, and Weiming Zhang. Stegaddpm: Generative image steganography based on denoising diffusion probabilistic model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7143–7151, 2023.

[33] Yinyin Peng, Yaofei Wang, Donghui Hu, Kejiang Chen, Xianjin Rong, and Weiming Zhang. Ldstega: Practical and robust generative image steganography based on latent diffusion models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3001–3009, 2024.

[34] Jiwen Yu, Xuanyu Zhang, Youmin Xu, and Jian Zhang. Cross: Diffusion model makes controllable, robust and secure image steganography. *Advances in Neural Information Processing Systems*, 36:80730–80743, 2023.

[35] Yiwei Yang, Zheyuan Liu, Jun Jia, Zhongpai Gao, Yunhao Li, Wei Sun, Xiaohong Liu, and Guangtao Zhai. Diffstega: towards universal training-free coverless image steganography with diffusion models. *arXiv preprint arXiv:2407.10459*, 2024.

[36] Qing Zhou, Ping Wei, Zhenxing Qian, Xinpeng Zhang, and Sheng Li. Improved generative steganography based on diffusion model. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[37] Xiaoxiao Hu, Sheng Li, Qichao Ying, Wanli Peng, Xinpeng Zhang, and Zhenxing Qian. Establishing robust generative image steganography via popular stable diffusion. *IEEE Transactions on Information Forensics and Security*, 2024.

[38] Daegyu Kim, Chaehun Shin, Jooyoung Choi, Dahuin Jung, and Sungroh Yoon. Diffusion-stego: Training-free diffusion generative steganography via message projection. *Information Sciences*, page 122358, 2025.

[39] Luke A Bauer, Wenxuan Bao, and Vincent Bindschaedler. Provably secure covert messaging using image-based diffusion processes. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 941–955. IEEE, 2025.

[40] Aqib Mahfuz, Georgia Channing, Mark van der Wilk, Philip Torr, Fabio Pizzati, and Christian Schroeder de Witt. Psyduck: Training-free steganography for latent diffusion. *arXiv preprint arXiv:2501.19172*, 2025.

[41] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[42] Xueying Mao, Xiaoxiao Hu, Wanli Peng, Zhenliang Gan, Zhenxing Qian, Xinpeng Zhang, and Sheng Li. From covert hiding to visual editing: robust generative video steganography. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 2757–2765, 2024.

[43] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the h. 264/avc video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7):560–576, 2003.

[44] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[45] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[46] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[47] Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. *arXiv preprint arXiv:2403.06098*, 2024.

[48] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

[49] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[50] Yun Cao, Hong Zhang, Xianfeng Zhao, and Xiaolei He. Steganalysis of h. 264/avc videos exploiting subtractive prediction error blocks. *IEEE Transactions on Information Forensics and Security*, 16:3326–3338, 2021.

[51] Rémi Cogranne, Vahid Sedighi, Jessica Fridrich, and Tomáš Pevnỳ. Is ensemble classifier needed for steganalysis in high-dimensional feature spaces? In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2015.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims presented in the abstract and introduction accurately reflect the paper's core contributions and overall scope, as supported by experimental results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Due to space constraints, we provide a detailed discussion of the limitations of our work in the supplementary material.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper does not include any formal theoretical assumptions, theorems, or proofs, and is therefore not applicable to this question.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides a detailed description of our method in the "Method" section and a comprehensive explanation of the experimental setup in the Experiments section. Additional implementation and configuration details are included in the supplementary material to ensure the reproducibility of the main results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We use the publicly available VidProM dataset [47]. The code used in our experiments is released at `https://anonymous.4open.science/r/LD-RoViS-7FB1/`, with full instructions to reproduce the main results. The code link has been anonymized.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The main paper provides detailed descriptions of the experimental setup, including data splits, training procedures, hyperparameter configurations, and other implementation details necessary to interpret the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For key experimental results, we report the average performance over 100 generated videos. We also include the standard deviation to quantify variability. This information is presented in the main paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We specify the computational environment in the experimental setup, including the use of four NVIDIA RTX A6000 GPUs. Additional details such as timesteps and computation times are reported in the supplementary material, ensuring reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research complies fully with the NeurIPS Code of Ethics. It does not involve any human subjects, sensitive data, or practices that may raise ethical concerns.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: This paper proposes a method for secure covert communication, which has potential positive applications in privacy protection and information security.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper does not release any models or datasets that pose potential risks of misuse, and thus no safeguards are necessary.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All the external datasets and models used in this work are properly cited in the paper. Their licenses and terms of use are clearly stated and have been fully respected.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new code assets introduced in this paper are thoroughly documented. Detailed instructions on the installation, usage, and replication of results are provided alongside the code to facilitate reproducibility.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Hyperparameter Tuning

In the LD-RoViS framework, the optimal values of three hyperparameters—τ_1, τ_2, and *cfg*—need to be determined through experiments. We generated stego videos using 10 different prompts and computed the average values of accuracy (acc), PSNR, BRISQUE, and capacity. The results are summarized in Table 6, 7, 8. To evaluate τ_1, we fixed $\tau_2 = 0.02$ and *cfg* = 16; to evaluate τ_2, we fixed $\tau_1 = 0.32$ and *cfg* = 16; and to evaluate *cfg*, we fixed $\tau_1 = 0.32$ and $\tau_2 = 0.02$.

We observe that as τ_1 and τ_2 increase, capacity increases, whereas acc, PSNR, and BRISQUE decrease. Similarly, as *cfg* increases, acc improves, PSNR and BRISQUE decrease, and capacity remains nearly unchanged. Our goal is to maximize capacity while maintaining acc $\geq 99\%$. On this basis, we choose the combination $\tau_1 = 0.32$, $\tau_2 = 0.02$, and *cfg* $= 16$. Notably, the BRISQUE score of the cover video is 28.83, which is very close to that of our stego videos. This finding indicates that although there are content differences between the stego and cover videos, their perceptual quality, as measured without reference, remains similar.

Table 6: Performance under different values of τ_1.

τ_1	acc(%)↑	PSNR(db)↑	BRISQUE↓	capacity(bits)↑
0.01	99.45	**43.11**	**28.80**	364
0.02	99.31	42.99	28.84	721
0.04	**99.52**	42.83	28.84	1464
0.08	99.42	42.56	28.83	2913
0.16	99.42	42.18	28.91	5869
0.32	99.17	41.66	28.90	11983
0.64	97.84	40.97	28.94	**24995**

Table 7: Performance under different values of τ_2.

τ_2	acc(%)↑	PSNR(db)↑	BRISQUE↓	capacity(bits)↑
0.01	**99.65**	**42.18**	**28.89**	5957
0.02	99.17	41.66	28.90	11983
0.04	98.53	41.01	28.96	23511
0.08	97.25	40.10	29.05	47708
0.16	94.80	39.02	29.25	100786
0.32	90.85	38.12	29.36	204572
0.64	82.90	37.57	29.48	**413237**

Table 8: Performance under different values of *cfg*.

cfg	acc(%)↑	PSNR(db)↑	BRISQUE↓	capacity(bits)↑
1	60.72	**43.15**	**28.83**	12168
2	68.56	43.04	28.84	**12206**
4	83.72	42.84	28.84	12083
8	95.45	42.43	28.79	12105
16	**99.17**	41.66	28.90	11983

B Validation on Other Video Diffusion Models

To further evaluate the generalization ability of the proposed LD-RoViS, we additionally implemented the method on another open-source text-to-video generation model, LTX-Video [48] (ltxv-2b-0.9.8-distilled version). LTX-Video is a two-stage diffusion model. The first stage focuses on generating high-quality image frames, while the second stage refines temporal consistency across frames. It

adopts a DiT-based architecture and employs an EDM sampler (non-deterministic). In our implementation, we retained the first stage unchanged and inserted the steganographic module at the final step of the second stage's denoising process. Due to architectural and sampling differences between Wan2.1 and LTX-Video, we re-tuned the hyperparameters using a small validation set and obtained the following optimal values:

$$k_s = 1.1, \quad \text{cfg} = 1.8, \quad \tau_1 = 0.32, \quad \tau_2 = 0.02.$$

Note that LTX-Video's generated video quality is slightly inferior to that of Wan2.1, which explains the higher BRISQUE scores reported in Table 9. In addition, the latent size in LTX-Video is approximately 2/5 of Wan2.1's latent size, leading to a corresponding reduction in embedding capacity. Experimental results demonstrate that LD-RoViS maintains stable performance under this different architecture and sampler, confirming the adaptability of our method.

Table 9: Performance comparison of LD-RoViS on different video generation models.

Model	Accuracy (%)↑	PSNR (dB)↑	BRISQUE ↓	Capacity (bits)↑
LD-RoViS (Wan2.1)	99.17	41.66	28.90	11983
LD-RoViS (LTX-Video)	99.23	41.39	34.21	4281

C Ablation on Modulation Time Step

To investigate the effect of the modulation time step in the CFG modulation strategy, we performed ablation studies by applying the modulation at different denoising steps. Here, t denotes the reverse denoising step counting from the last step, *i.e.*, $t = 2$ corresponds to the second-to-last denoising step. The results are shown in Table 10. When the modulation is applied at earlier denoising steps, the resulting latent variables X_0^1 and X_0^2 become more uncontrollable. After several denoising steps, the difference between X_0^1 and X_0^2 may either increase or decrease unpredictably. As our message extraction relies heavily on the distinguishability between X_0^1 and X_0^2, this uncertainty leads to lower extraction accuracy with higher standard deviations. Meanwhile, PSNR, BRISQUE, and steganalysis results remain nearly unchanged, indicating that perceptual quality and security are preserved. Therefore, the final-step modulation is adopted in our design.

Table 10: Ablation study on different modulation time steps.

Time step t	Accuracy (%)↑	PSNR (dB)↑	BRISQUE ↓	CovNet P_e(%)↑
t=1 (Ours)	99.17±0.63	41.66	28.90	49.74
t=2	98.43±1.12	41.22	29.99	49.93
t=4	95.72±2.57	41.34	28.06	49.78
t=8	94.82±3.30	41.94	29.62	49.72

D Time Consumption

The embedding time is also a critical metric for evaluating steganographic methods. Methods with lower embedding times are generally more practical for real-world applications. To assess this, we measured the embedding latency using Python's `time` package, and the results are shown in Table 11. Notably, since LD-RoViS performs steganographic embedding during the video generation process, the actual embedding time should be calculated by subtracting the time required to generate a clean video via the video model. The experimental results show that LD-RoViS introduces minimal time overhead (30.48 s), significantly outperforms AQIM and MEC_AQIM, and achieves a runtime comparable to that of RoGVSN.

E More Experimental Details

In this section, we provide more experimental details to ensure the reproducibility of LD-RoViS. We adopt the T2V-1.3B model from Wan2.1 [30] as the video generation model. This model has

Table 11: Embedding latency (in seconds) for different methods.

Method	Video Model	LD-RoViS	RoGVSN	AQIM	MEC_AQIM
Additional Time (s)↓	366.02	30.48	36.52	698.80	697.59

low memory requirements and can run on a single RTX 4090 Ti. It also offers fast generation speed, producing a video in just 50 time steps. On average, it takes approximately 6 minutes to generate an 81-frame video at a resolution of 480×832. For steganographic tasks, the average generation time is approximately 7 minutes per video.

Dataset: Our experiments use a dataset consisting of 100 cover videos and 100 stego videos. The prompts for the cover and stego videos are exactly the same. The cover videos are generated via a clean version of the Wan2.1 model, whereas the stego videos are produced via a Wan2.1 model with the steganographic module. All prompts are randomly selected from the VidProM dataset [47]. However, some prompts from VidProM contain parameters unsupported by Wan2.1. We manually filtered out such prompts. Table 12 lists a subset of the prompts used in our experiments.

Table 12: Example prompts used for video generation.

1	a kangaroo dancing to electronic music in a crowded nightclub.
2	a lady wandering in the enchanted woods, lost and confused.
3	aliens walking in city.
4	cars on the beach are attacked by sharks.
5	Animals working together, sharing resources, and demonstrating cooperation under the guidance of Leo, the wise lion.
6	The black Chinese dragon opens its eyes.
7	a wolf walking in the jungle.
8	A massive storm hitting a city.
9	a wild cat running in the jungle.
10	an elephant riding a bike, masterpiece, cinematic.

F More Visual Results

In this section, we present more visual results to evaluate the impact of steganographic embedding on video quality.

Visual Quality across Video Frames: A common challenge in traditional video steganography is distortion drift, where embedding distortions gradually propagate across frames, resulting in noticeable artifacts in later parts of the video. To investigate whether LD-RoViS suffers from a similar issue, we decomposed four generated stego videos into individual frames via FFmpeg. We then visualized the first frame, middle frame (41st), and last frame (81st) in Fig. 4. The results show that the steganographic modifications made in the latent space by LD-RoViS do not introduce visible distortions or artifacts. The visual quality remains stable throughout the video.

Pixel-Wise Differences between Cover and Stego Videos: In the main paper, Fig. 3 shows that the pixel differences between cover and stego videos generated by LD-RoViS are minimal and are mostly concentrated in regions with complex textures. In this section, we provide a deeper analysis of these results. Fig. 5 presents additional examples of pixel-level differences. The "diff" images represent the absolute pixel difference between the 41st frame of the cover and stego videos, with brightness amplified by a factor of 10 for better visibility. These visualizations confirm that pixel differences are extremely subtle and imperceptible to the human eye. Notably, most differences appear along object contours. We hypothesize that this is due to the low weighting of the predicted noise at the final step of the diffusion model's reverse process. Even large modifications to this predicted noise have minimal effects on the final output. This suggests that the last step of the diffusion process provides

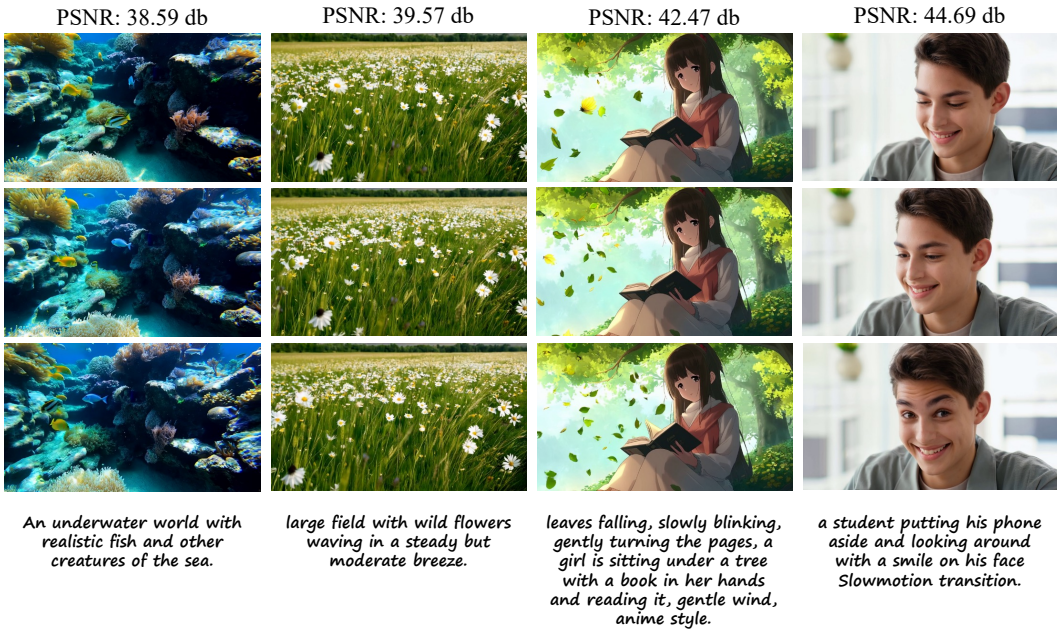

Figure 4: The first frame, middle frame (41st), and last frame (81st) of a stego video generated by LD-RoViS, along with their corresponding prompts.

a naturally robust channel for steganography—allowing hidden information to be embedded with negligible visual degradation and strong resistance to detection.

G Limitations

Although the proposed LD-RoViS enables robust video steganography and outperforms existing methods in terms of both security and visual quality, it also has certain limitations.

The quality of the stego video depends heavily on the performance of the video generation model. While video generation models are rapidly improving, they still occasionally produce low-quality outputs. As shown in Fig 6, some generated videos exhibit physically implausible deformations, artifacts, or distorted faces. These low-quality samples may raise suspicion during transmission between the sender and receiver, thereby reducing the practical usability of the steganographic method.

LD-RoViS inevitably alters the video generation process. Specifically, it modifies the parameters at the final timestep of the reverse diffusion process, which may introduce slight shifts in the distribution of the generated videos and lead to a decrease in visual quality. However, the experimental results show that steganalysis tools fail to detect meaningful distribution differences between the cover and stego videos, indicating that LD-RoViS remains secure against detection. Additionally, the stego videos maintain high PSNR and favorable BRISQUE scores, suggesting that LD-RoViS achieves strong practicality despite the minimal distortion introduced.

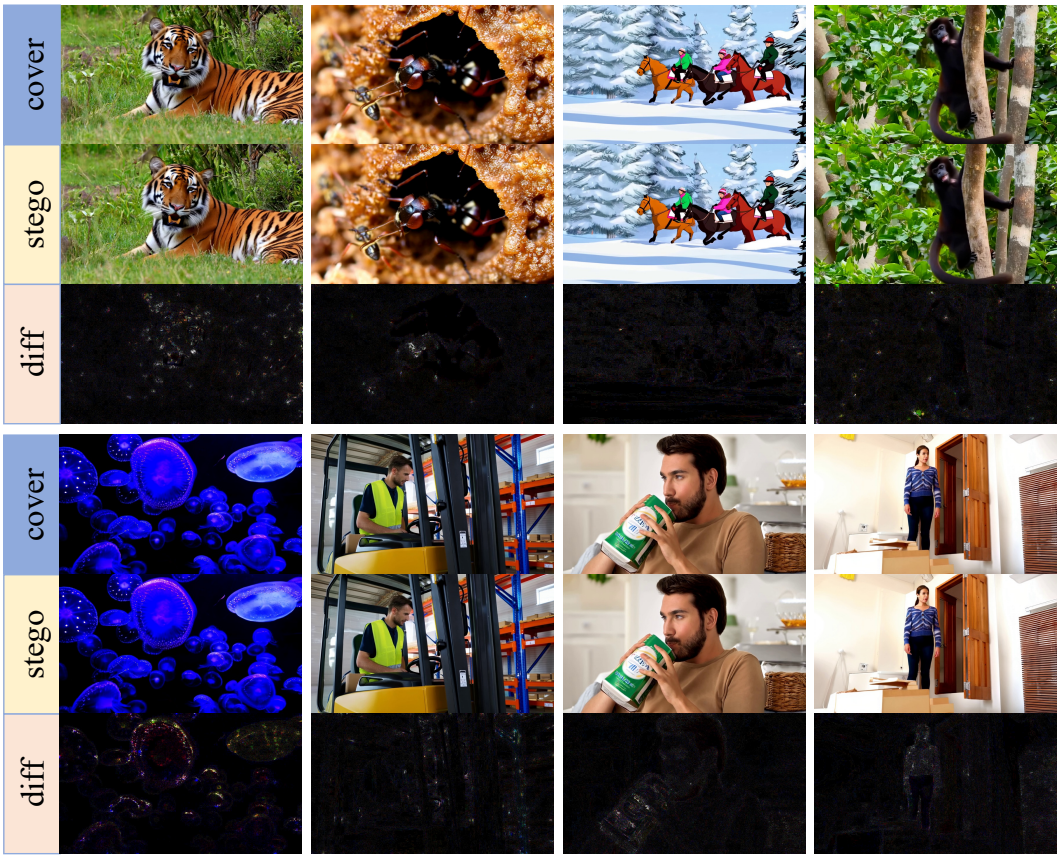

Figure 5: Cover video, stego video, and their pixel-wise difference ("diff" denotes the pixel difference, with brightness amplified by a factor of 10 for better visibility).

Figure 6: Examples of low-quality generated videos, where noticeable distortions and artifacts are present in the frames.