2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

BDTIAG: Reliable and Efficient Black-Box Adversarial Text-to-Image Generation via Decision Boundary Exploration

Yongqi Jiao[1], Yucheng Shi[2], Yufei Gao[1], Lin Wei[1], and Lei Shi[1(✉)]

[1] School of Cyber Science and Engineering, Zhengzhou University, China
[2] School of Computer and Artifcial Intelligence, Zhengzhou University, China
jyq917@gs.zzu.edu.cn

Abstract. Text-to-image generation models can produce high-quality images from textual descriptions. However, they are vulnerable to adversarial attacks, which can manipulate outputs and bypass content moderation systems, leading to potential security risks. We propose BDTIAG, a black-box adversarial attack framework that improves attack efficiency and stealthiness. It comprises two key phases: (1) Adversarial Sample Space Expansion (ASSE), which systematically perturbs text to generate diverse adversarial samples, and (2) Boundary Perturbation Backtracking (BPB), which refines these samples to maximize attack sccess while minimizing detection. Extensive experiments on DALL·E, DALL·E 2, Imagen, and AttnGAN demonstrate that BDTIAG outperforms existing black-box attack methods, achieving a **6.25%** increase in attack success rate and reducing the number of queries by **41.02%** compared to RIATIG, all while preserving semantic consistency and naturalness.

Keywords: Adversarial Attack, Text-to-Image Generation, Black-Box Framework, Information Security, Semantic Perturbation

1 Introduction

Recent advances in deep learning, particularly in diffusion and transformer-based models, have enabled text-to-image generation systems to produce high-resolution, semantically coherent images from natural language descriptions. State-of-the-art models such as DALL·E, DALL·E 2, and Imagen [15,16,17] leverage large-scale training datasets and multimodal representations to translate textual prompts into photorealistic images with remarkable semantic accuracy. Adversarial attacks against text-to-image models generally fall into two categories: (1) untargeted attacks, which introduce textual perturbations to induce unpredictable or unintended outputs, and (2) targeted attacks, which craft adversarial prompts to generate specific images while bypassing moderation systems, potentially leading to misinformation or harmful content generation. Some standard techniques [6,7] used in these attacks include exploiting hidden vocabularies, using morphological similarity between words, and replacing characters with visually similar ones. However, existing adversarial attacks suffer from several limitations. Many methods require an excessive number of queries to effectively perturb text, making them computationally expensive and impractical for real-world applications where API-based models impose query restrictions. Furthermore, adversarial prompts generated by current methods often degrade linguistic quality, resulting in ungrammatical, incoherent, or semantically ambiguous text, which can hinder attack effectiveness.

To address these challenges, we introduce Black-box Decision-boundary-based Text-to-Image Adversarial Generation (**BDTIAG**). Unlike prior methods, BDTIAG exploits decision-boundary information from text-to-image models, enabling the generation of adversarial prompts with minimal queries while maintaining high stealthiness and naturalness (Figure 1). BDTIAG consists of two steps: Adversarial Sample Space Expansion (**ASSE**) and Boundary Perturbation Backtracking (**BPB**) and the boundary perturbation concept is illustrated in Figure 3. The contributions of this work are summarized as follows:

High-efficiency. BDTIAG is designed to be highly efficient in black-box settings. Our method leverages ASSE and BPB to generate adversarial examples efficiently. By reducing the required queries, BDTIAG achieves high attack success rates with fewer computational costs. This makes it more practical for real-world attack scenarios. For instance, on the DALL·E model, BDTIAG achieves an attack success rate (ASR) of **86.25%** compared to RIATIG's **83.75%**, while reducing the average number of queries (NoQ) by **49.3%** (from 2843.64 to 1440.73). Similar improvements are observed across other models: for AttnGAN, BDTIAG reduces NoQ by **47.7%** (from 2652.03 to 1386.24) while increasing ASR to **90.00%** (vs. RIATIG's 86.25%).

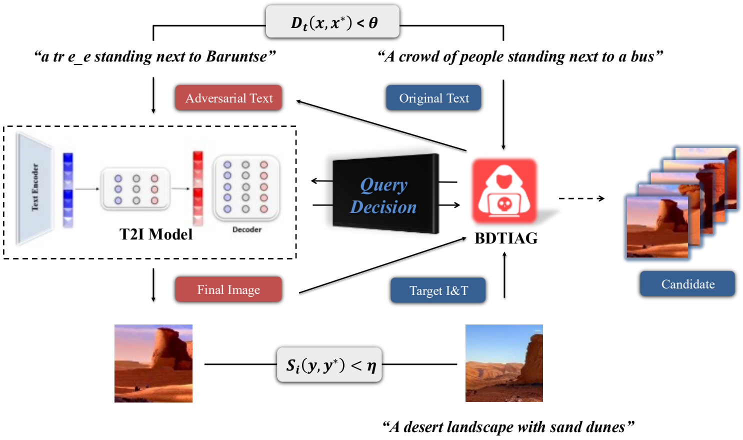

Fig. 1. BDTIAG obtains information by querying the target model to get the adversarial text.

High-quality and stealthy adversarial samples. BDTIAG focuses on generating high-quality adversarial samples that are both natural-looking and semantically coherent. The ASSE stage uses advanced natural language processing techniques to expand the sample space while maintaining semantic and syntactic integrity. The BPB stage then refines these samples to ensure they are close to the target image and have a reasonable semantic distance from the original text, making them difficult to detect. Experimental results highlight that BDTIAG generates adversarial texts with lower perplexity (PPL) and competitive semantic distance (Dist). For example, on AttnGAN, BDTIAG achieves a PPL of **531.77** (vs. RIATIG's 562.67). Furthermore, robustness tests show that adversarial samples generated by BDTIAG maintain an **89%** ASR even after repeated inputs to the target model (Table 3), underscoring their stability and stealthiness.

Comprehensive evaluation and contribution to security awareness. We conduct extensive evaluations of BDTIAG against multiple well-known Text-to-Image generation models. By demonstrating the effectiveness of BDTIAG, we aim to raise

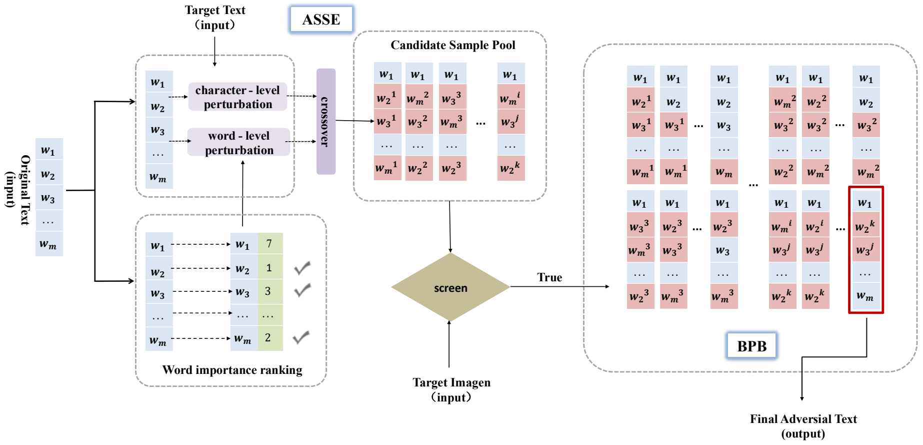

Fig. 2. An overview of the proposed BDTIAG. The blue blocks show the original words, and the pink blocks represent the substituted words.

awareness of the security vulnerabilities in these models. Our findings underscore the need for the research community to develop more robust defenses for Text-to-Image generation systems.

2 Related Work

Text-to-image generation has evolved significantly, starting with early GAN-based models like AttnGAN [11], which improved text-image alignment but faced diversity limitations. Subsequent models like DFGAN and DMGAN [12,13] built on this foundation, while OpenAI's DALL·E series—evolving from DALL·E to DALL·E 2, integrated with ChatGPT—marked a significant advancement with enhanced outputs. Google's Imagen and Stable Diffusion [14] further advanced the field with innovative attacks that have emerged as a critical concern. Untargeted attacks, like Daras and Dimakis' exploitation of hidden vocabulary [7], disrupt model outputs to test robustness, while targeted attacks, such as RIATIG's [1] sentence-level tweaks or Millière's fabricated word techniques, aim to produce specific images while bypassing security filters. These attacks reflect a core challenge: for a text-to-image model mapping text T to images I(denoted $G: T \rightarrow I$), an attacker crafts an adversarial prompt x^* from an original x, aiming to generate an image $G(x^*)$ close to a target y_t in meaning (judged by image similarity) while keeping x^* sufficiently distinct from x.

3 Methodology

3.1 Attack Overview

Before delving into the detailed design, we present an overview of BDTIAG. As shown in Figure 2, BDTIAG mainly consists of two steps: Adversarial Sample Space Expansion (ASSE)and Perturbation Backtracking (BPB). In the ASSE phase, the original text undergoes character-level and semantic-level perturbations, combined with crossover operations, to quickly generate a diverse pool of adversarial samples capable of producing images similar to the target image, regardless of the generated text's quality. In the BPB phase, these samples are systematically optimized by iteratively reversing perturbations and evaluating their effectiveness using image similarity S_i and semantic distance D_t. The final adversarial prompt is selected based on a weighted score that balances attack effectiveness and stealthiness, ensuring minimal queries to the target model while achieving high-quality adversarial outputs.

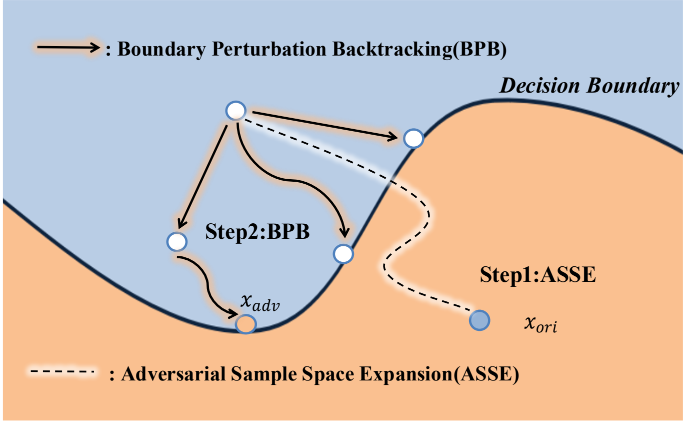
: Boundary Perturbation Backtracking(BPB)
Decision Boundary
Step2:BPB
Step1:ASSE
x_{adv}
x_{ori}
: Adversarial Sample Space Expansion(ASSE)

Fig. 3. The BDTIAG crosses the decision image boundary in step 1 to obtain the candidate samples X_{ASSE} , then finds x_{adv} closer to the x_{ori} one in step 2.

3.2 Similarity Measurement

Selecting suitable metrics requires evaluating visual space coherence and image semantic representation. CLIP model is trained on many image-text pair data and can effectively align visual and text information. Therefore, in this paper, the pre-trained image encoder of CLIP is used to encode images, and the cosine distance of the encoded vectors is calculated to measure the similarity of images [2]. For image, the formula is:

$$S_i(G(x), y) = \frac{E_i(G(x)) \cdot E_i(y)}{\| E_i(G(x)) \| \| E_i(y) \|} \tag{1}$$

For text, the formula is:

$$D_t(x, x^*) = 1 - \frac{E_t(x) \cdot E_t(x^*)}{\| E_t(x) \| \| E_t(x^*) \|} \tag{2}$$

where S_i(the semantic similarity of images) and D_t (the semantic distance of texts) are computed using pre-trained CLIP image encoder $E_i(\cdot)$ and BERT text encoder $E_t(\cdot)$.

3.3 Adversarial Sample Space Expansion (ASSE)

Using the target text as a reference, the ASSE applies extensive mutations to the original text. The primary purpose of ASSE is to generate an initial pool of adversarial texts that can guide Text-to-Image models to produce images similar to the target images while being semantically distinct from the original text used to generate the target images.

Ranking the Importance of Words. The method ranks word importance as follows: Initially, a pre-trained BERT model [8] and its tokenizer process the input sentence by tokenizing it and adding unique tokens. Then, for each word, replace it with a mask token. Next, the BERT model predicts the word at the mask position from context, yielding a probability distribution. Based on information theory (low-probability events carry more info), a word's importance score is calculated as one minus the predicted probability of the original word in this distribution. Thus, lower predicted probabilities mean higher scores and more outstanding semantic contributions. Finally, words are ranked by the score in descending order to show their importance in the sentence.

Preliminary Perturbation. For these crucial words which play a decisive role in sentence semantics, we carry out the initial perturbation from three methods: (1) character-level perturbation, insert spaces or underscores within words (e.g., “bench” to “ben c h”); randomly swap middle characters without changing the first and last ones (e.g., “umbrella” to “umbrela”); remove a non-initial and non-final character (e.g., “sitting” to “sitin”). (2) Intelligent perturbation based on semantic similarity, Leverage pretrained word vector model(Word2Vec[9]) to find semantically similar words in the vector space for key elements. (3) Crossover[5] inspired by the principles of biological inheritance, where genetic recombination drives diversity in offspring, we devise an advanced crossover strategy to synthesize novel adversarial samples by blending semantic traits from existing ones. This process begins by randomly selecting two parent samples from the current adversarial pool and leveraging a pre-trained BERT model. We identify key semantic traits within each sample, which act as the inheritable units akin to genes in biology. The crossover then prioritizes exchanging traits with higher importance scores, ensuring that dominant elements are recombined.

Candidate Samples Screening. After generating the candidate samples, the algorithm filters them based on their semantic distance from the original text x_t. Samples with a semantic distance $D_t(x^*, x_t)$ above the threshold are retained. The algorithm then evaluates the similarity between the generated images and the target image y_t using the CLIP model. Only samples that meet the similarity threshold are added to the final set of adversarial samples S_{adv}.

Algorithm 1 Boundary Perturbation Backtracking (BPB)

```
Input: Original Text $x_t$, Adversarial samples $S_{adv}$, Target Image $y_t$, Perturbation Records $R$, $\eta$, $\theta$, $\alpha$, $l$
Output: Adversarial sample $x^*$
$x^* \leftarrow \emptyset$, best_score ← 0
for each $x^* \in S_{adv}$ do
    $O \leftarrow R[x^*]$
    for each backtracking path $p \in$ generate_paths ($O$) do
        $x^*_{modified} \leftarrow x^*$
        backtracking_count ← 0
        for each operation $o_i \in p$ do
            if backtracking_count $\geq l$ then
                break
            end if
            $x^*_{modified}$ ← reverse_perturbation ($x^*_{modified}$, $o_i$)
            PPL ← Perplexity ($x^*_{modified}$)
            if PPL > 1200 then
                continue
            end if
            backtracking_count ← backtracking_count + 1
        end for
        $y^* \leftarrow G(x^*_{modified})$
        score ← $\alpha \cdot S_i(y^*, y_t)$ + (1-α) · $D_t(x^*_{modified}, x_t)$
        if $S_i(y^*, y_t) \geq \eta$ and $D_t(x^*_{modified}, x_t) \geq \theta$ and score > best_score then
            $x^* \leftarrow x^*_{modified}$
            best_score ← score
        end if
    end for
end for
return $x^*$
```

3.4 Boundary Perturbation Backtracking (BPB)

The scoring function evaluates the quality of adversarial samples by balancing the tradeoff between image similarity and semantic distance. The modified scoring function is defined as:

$$Score(x^*) = \alpha \cdot S_i(G(x^*), y_t) + (1-\alpha) \cdot D_t(x^*, x_t) \tag{3}$$

where α is a weighting factor that controls the tradeoff between image similarity and semantic distance. The workflow of BPB is presented in Algorithm 1. The key steps of BPB are as follows:

Backtracking Perturbation Operations. During the ASSE phase, all perturbation operations applied to generate adversarial samples are recorded (O). In the BPB stage, these operations are systematically reversed in a controlled manner. For each adversarial sample, the algorithm explores multiple backtracking paths by reversing subsets of the recorded perturbations. Specifically, perturbations are prioritized based on their impact on semantic distance—operations that caused the largest semantic deviations are reversed first. This strategy aims to retain necessary modifications for attack success while maximizing stealthiness.

Dynamic Evaluation and Filtering. After each backtracking step, the modified text $x^*_{modified}$ is evaluated for naturalness using perplexity (PPL). If the PPL is greater than the threshold, the sample is discarded to ensure text fluency. The refined text is then fed into the target model to generate an image y^*.

Multi-Path Optimization. To balance exploration and efficiency, the algorithm adopts a multi-path backtracking strategy. It generates diverse paths by reversing different combinations of perturbations and retains only those samples that satisfy predefined thresholds (η for similarity and θ for semantic distance). The number of cycles of this process is controlled by the maximum backtracking step l, The final adversarial sample x^* is selected based on the highest weighted score, which harmonizes attack effectiveness (S_i) and stealthiness (D_t).

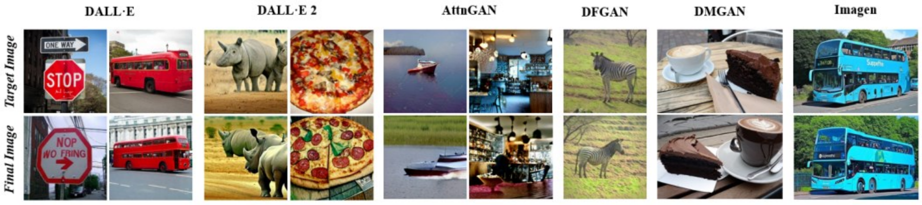

Fig. 4. Part of the target image and the adversarial image

4 Experiments

4.1 Experiments Settings

Datasets and target models. We choose the COCO 2014 dataset [3] with 82,783 training and 40,504 validation images, each having five text descriptions and belonging to 80 categories. To ensure reliable and comprehensive results in sampling, we randomly selected one image-text pair per category as the target and another text as the original. This formed 80 experimental sample groups; we target models with various architectures: GAN-based AttnGAN, DFGAN, DMGAN; Transformer-based DALL·E 1; diffusion-based Imagen; and DALL·E 2 (Transformer-diffusion hybrid). Part of the target image and the adversarial image obtained is shown in Figure 4.

Baselines. To assess BDTIAG's black-box performance, we chose baseline methods for targeted and untargeted attacks. For targeted attacks, RIATIG (designed for Text-to-Image models) allows direct comparison with BDTIAG in creating adversarial samples like specific images, highlighting BDTIAG's advantages. For untargeted attacks, we included MacPromp [6], EvoPromp [6], and TextFooler [4].

Parameter Analysis. According to the parameter analysis results in Table 1, We set the maximum cross-sample number (M) and the weight factor (α) to 50 and 0.7.

Table 1. (a) Maximum cross-sample number (M) and corresponding ASR in ASSE step (b) Weight factor(α)and corresponding ASR in BPB step

M	**30**	**35**	**40**	**45**	**50**	**55**	**60**	**65**	**70**
DALL·E	1/10	2/10	4/10	8/10	**9/10**	9/10	9/10	9/10	9/10
DALL·E 2	1/10	3/10	3/10	5/10	**7/10**	8/10	8/10	8/10	8/10
Imagen	0/10	2/10	3/10	5/10	**8/10**	8/10	8/10	8/10	8/10
AttnGAN	2/10	4/10	5/10	7/10	**9/10**	9/10	9/10	9/10	10/10

(a)

α	**0.00**	**0.05**	**0.10**	**0.15**	**0.20**	**0.25**	**0.30**	**0.35**	**0.40**	**0.45**	**0.50**
DALL·E	5/10	5/10	7/10	8/10	8/10	8/10	**9/10**	9/10	9/10	9/10	9/10
DALL·E 2	3/10	3/10	4/10	5/10	5/10	7/10	**7/10**	8/10	8/10	8/10	8/10
Imagen	3/10	5/10	6/10	6/10	7/10	7/10	**8/10**	8/10	8/10	8/10	8/10
AttnGAN	3/10	3/10	4/10	5/10	6/10	6/10	7/10	**8/10**	8/10	8/10	8/10

(b)

Note: The notation (e.g., 9/10) indicates the number of successful adversarial samples generated out of 10 total experiments, where the numerator represents successes and the denominator represents total trials.

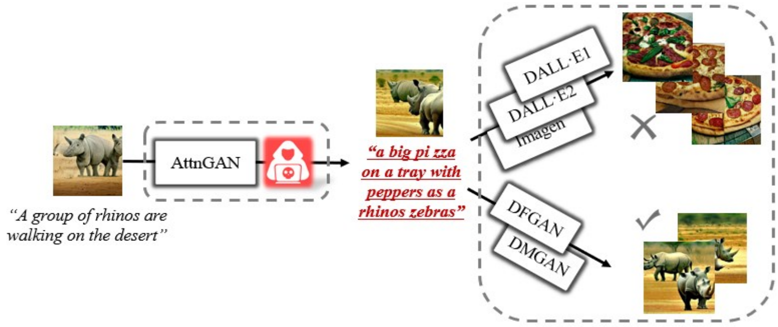
"A group of rhinos are walking on the desert"
AttnGAN
"a big pi zza on a tray with peppers as a rhinos zebras"
DALL·E1
DALL·E2
Imagen
×
DFGAN
DMGAN
✓

Fig. 5. The adversarial text "a big pi zza on a tray with peppers as a rhinos zebras" from AttnGAN is input to other models, DFGAN, DMGAN, like AttnGAN, outputs a picture of rhinos, while DALL·E, DALL·E 2, and Imagen outputs a picture of pizza

4.2 Evaluation Metrics

As indicated by references, we utilize R-precision, a prevalent metric in this field. This metric assesses the semantic alignment between the generated image and the provided text description. We specifically focus on the case when R is one and define a result of True for R-1 as a successful attack. We use Perplexity (**PPL**) to measure the naturalness of text. The lower the PPL, the higher the concealment of the attack. We use cosine similarity score (**Dist**) to measure semantic distance [10], and the lower the value, the more similar the semantics of the text. The Average Number of Queries (**NoQ**) measures the efficiency of the adversarial attack by calculating the average number of queries required to generate a successful adversarial text.

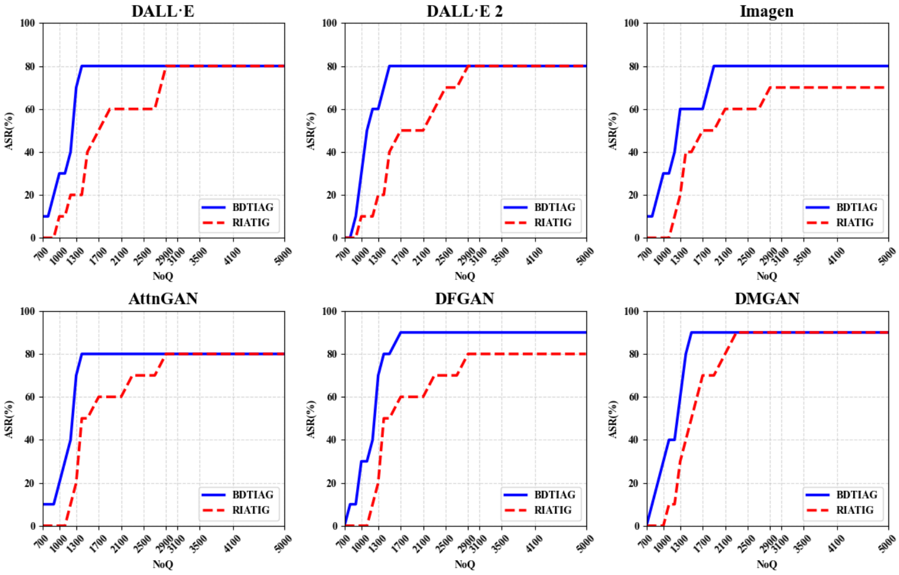

Fig. 6. Tradeoff between Number of Queries (NoQ) and Attack Success Rate (ASR) for BDTIAG and RIATIG across Multiple Target Models

4.3 Experimental Results

Evaluation of Attack Performance. This study evaluates BDTIAG's attack performance against Text-to-Image models (DALL·E, DALL·E 2, AttnGAN, DFGAN, DMGAN, Imagen) compared to baselines (TextFooler, MacPromp, EvoPromp, RIATIG). Table 7 shows BDTIAG excels in ASR (e.g., 86.25% on DALL·E, 82.50% on DALL·E 2, 90.00% on AttnGAN), with low Dist (e.g., 0.26 on DFGAN vs. RIATIG's 0.25) and competitive PPL, reflecting coherence. It also reduces NoQ significantly below RIATIG's average, proving its superior effectiveness and efficiency in black-box adversarial generation.

Tradeoff Between NoQ and ASR. Figure 6 shows the NoQ-ASR tradeoff. Overall, for all target models, both methods' ASR rises with NoQ, and BDTIAG's ASR grows faster. For most models at similar ASR, BDTIAG needs fewer NoQ.

Transferability. We used the adversarial examples generated for one model to attack other models. Figure 5 shows the experimental process of one of the example adversarial samples. The experimental results in Table 2 demonstrate that BDTIAG's ASR is 30.83%, while RIATIG's can only reach 22.50%. Therefore, our method has better portability.

Adversarial Sample Robustness. We fed the exact adversarial text to the target model ten times (like Figure 7) and tested the ASR. The experimental results (Table 3) show that the success rate reaches 89%, indicating that the adversarial sample exhibits a certain level of robustness.

Table 2. Transferability of BDTIAG. Row *i* is the model used to generate adversarial samples, and column *j* is the target model.

	DALL·E	**DALL·E 2**	**Imagen**	**AttnGAN**
DALL·E	-	4/10	1/10	2/10
DALL·E 2	2/10	-	6/10	1/10
Imagen	2/10	6/10	-	2/10
AttnGAN	3/10	4/10	4/10	-

Word Importance Ranking Method. As shown in Figure 8, experimental results compare an alternative method and ours. The alternative model's score has a single peak and is overall flatter. Ours has at least two peaks, making capturing multiple important words' contributions better. Thus, our method identifies text key info better.

Table 3. Feed the exact adv text to the target model ten times and calculate the ASR

AttnGAN	**DFGAN**	**DMGAN**	**DALL·E**	**DALL·E 2**	**Imagen**
8/10	9/10	9/10	9/10	10/10	10/10

Influence of Datasets. Ten text-image pairs were randomly selected from coco2014, 2017, and Flickr30k as attack targets to evaluate each index. The experimental data in Table 4 shows our method performed stable on different datasets.

4.4 Ablation Study

Adversarial Sample Space Expansion. In the ablation study of ASSE (Table 6), single character-level perturbation and its combination with the crossover operation have low attack success rates and struggle to launch successful attacks. Semantic perturbation alone increases the attack success rate to some extent, and combining it with the crossover operation can further enhance this rate. Integrating character-level perturbation, semantic perturbation, and the crossover operation simultaneously can significantly boost the attack success rates across different models, indicating that the synergistic effect of multiple perturbation strategies is crucial for BDTIAG.

Boundary Perturbation Backtracking. We did an ablation experiment to validate the importance of the BPB phase in BDTIAG, removing BPB and using only ASSE-generated samples for the attack. BPB refines samples by backtracking perturbations to boost attack success and stealth. As depicted in Table 5, BPB is crucial for optimizing samples, enhancing attack, and ensuring stealth.

Table 4. Impact of Different Datasets on BDTIAG Attack Effect across AttnGAN, DALL·E, and DALL·E 2 Based on ASR, PPL, Dist, and NoQ Indicators

Datasets	AttnGAN				DALL·E				DALL·E 2			
	ASR	PPL	Dist	NoQ	ASR	PPL	Dist	NoQ	ASR	PPL	Dist	NoQ
coco2014	9/10	570.35	0.25	1470.1	8/10	663.73	0.29	1598.4	8/10	529.62	0.28	1588.2
coco2017	9/10	603.28	0.24	1525.9	8/10	547.90	0.25	1677.2	8/10	563.97	0.28	1729.9
Flickr30r	9/10	584.11	0.22	1503.3	9/10	637.89	0.25	1706.6	8/10	603.97	0.28	1680.5

Table 5. Impact of BPB Phase on Adversarial Attack Performance across AttnGAN, DALL·E, and Imagen Models

Ablation Study	AttnGAN			DALL·E			Imagen		
	ASR↑	PPL↓	Dist↓	ASR↑	PPL↓	Dist↓	ASR↑	PPL↓	Dist↓
w/o BPB	1/10	1532.63	0.41	0/10	1487.75	0.39	0/10	1579.62	0.44
w/ BPB	10/10	642.35	0.24	9/10	577.91	0.25	9/10	523.97	0.28

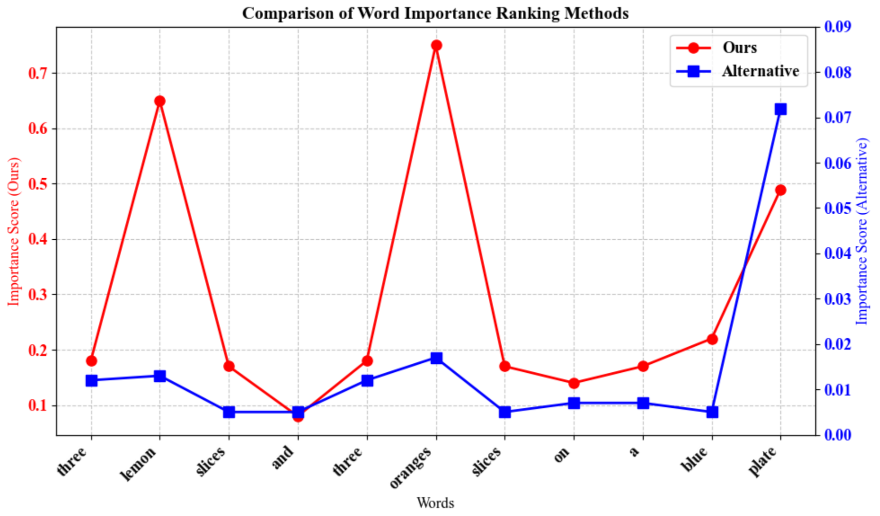

Fig. 8. Selecting essential words for the example text

Table 6. Results of Ablation Study on ASSE, “char” represents character-level perturbation, “w2v” represents semantic perturbation, and “cr” represents crossover

Ablation Study			**AttnGAN**			**DALL·E**			**Imagen**		
char	**w2v**	**cr**	**ASR↑**	**PPL↓**	**Dist↓**	**ASR↑**	**PPL↓**	**Dist↓**	**ASR↑**	**PPL↓**	**Dist↓**
√			0/10	673.1	0.14	0/10	618.7	0.11	0/10	629.6	0.13
√		√	0/10	973.5	0.32	0/10	848.1	0.29	0/10	901.5	0.35
	√		1/10	227.3	0.19	0/10	263.9	0.15	0/10	244.7	0.16
	√	√	3/10	723.1	0.29	2/10	697.2	0.27	1/10	703.6	0.21
√	√	√	9/10	582.5	0.25	8/10	627.3	0.24	8/10	524.9	0.26

Influence of Original Text. We conducted 10 training sessions using different original texts for the exact target text and image. The results in Table 8 demonstrate that the BDTIAG attack exhibits strong robustness against different original text attacks.

Table 7. The results of the BDTIAG’s attack performance

Model	**Method**	**ASR↑**	**Dist↓**	**PPL↓**	**NoQ↓**
DALL·E	TextFooler	0.02%	0.23	1838.09	
	MacPromp	66.25%	0.43	5866.41	-
	Evopromp	57.50%	0.16	5527.78	-
	RIATIG	83.75%	0.26	722.36	2843.64
	Ours	**86.25%**	0.24	647.03	**1440.73**
DALL·E 2	TextFooler	0.00%	0.24	1993.95	-
	MacPromp	68.75%	0.42	4297.33	-
	Evopromp	67.42%	0.14	5183.71	-
	RIATIG	78.75%	0.27	883.09	2652.03
	Ours	**82.50%**	0.24	787.82	**1467.60**
AttnGAN	TextFooler	0.05%	0.18	1706.86	-
	MacPromp	43.75%	0.50	5267.29	-
	Evopromp	51.25%	0.14	5084.48	-
	RIATIG	86.25%	0.25	562.67	2652.03
	Ours	**90.00%**	0.24	531.77	**1386.24**
DFGAN	TextFooler	0.75%	0.16	1829.23	-
	MacPromp	55.28%	0.48	5298.17	-
	Evopromp	48.75%	0.17	4397.75	-
	RIATIG	82.50%	0.25	499.13	2826.75
	Ours	**88.75%**	0.26	492.93	**1255.68**
DMGAN	TextFooler	0.05%	0.17	1757.97	-
	MacPromp	59.03%	0.43	5527.29	-
	Evopromp	44.39%	0.14	5220.18	-
	RIATIG	82.50%	0.24	505.98	2751.99
	Ours	**88.75%**	0.23	508.27	**1295.41**
Imagen	TextFooler	0.00%	0.18	1807.75	-
	MacPromp	56.25%	0.43	5339.94	-
	Evopromp	47.50%	0.19	5220.18	-
	RIATIG	86.25%	0.26	452.26	2834.96
	Ours	86.25%	0.26	511.32	**1643.91**

Table 8. Evaluation of BDTIAG Attacks' Robustness to Different Original Texts

	ASR↑	Dist↓	PPL↓	NoQ↓
AttnGAN	9.1(±0.18)/10	0.24±0.04	596.33±43.75	1414.21±203.68
DALL·E	8.8(±0.22)/10	0.24±0.03	475.75±53.02	1639.57±180.56
Imagen	8.4(±0.17)/10	0.26±0.04	470.83±50.53	1501.08±187.76

5 Conclusion

We propose BDTIAG, a black-box adversarial attack framework for text-to-image generation that integrates Adversarial Sample Space Expansion (ASSE) and Boundary Perturbation Backtracking (BPB) to enhance attack efficiency and stealth. Extensive experiments demonstrate BDTIAG's superiority, achieving up to **6.25%** higher attack success rates and **41.02%** fewer queries compared to state-of-the-art methods while preserving semantic coherence. This work highlights critical security vulnerabilities in text-to-image models and calls for robust defense mechanisms. Future research will focus on unified defense strategies against multimodal adversarial attacks and improving model resilience.

References

1. Liu H, Wu Y, Zhai S, et al. Rating: Reliable and imperceptible adversarial text-to-image generation with natural prompts[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 20585-20594
2. Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision[C]//International conference on machine learning. PmLR, 2021: 8748-8763.
3. Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//Computer vision–ECCV 2014: 13th European conference, Zurich, Switzerland, September 6-12, 2014, proceedings, part v 13. Springer International Publishing, 2014: 740-755.
4. Jin D, Jin Z, Zhou J T, et al. Is Bert really robust? A strong baseline for natural language attack on text classification and entailment[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(05): 8018-8025.
5. Holland J H. Genetic algorithms[J]. Scientific American, 1992, 267(1): 66-73.
6. Millière R. Adversarial attacks on image generation with made-up words[J]. arXiv preprint arXiv:2208.04135, 2022.
7. Daras G, Dimakis A G. Discovering the hidden vocabulary of dalle-2[J]. arXiv preprint arXiv:2206.00169, 2022.
8. Tsai H, Riesa J, Johnson M, et al. Small and practical BERT models for sequence labeling[J]. arXiv preprint arXiv:1909.00100, 2019.
9. Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.
10. Cer D, Yang Y, Kong S, et al. Universal sentence encoder[J]. arXiv preprint arXiv:1803.11175, 2018.

11. Xu T, Zhang P, Huang Q, et al. Attngan: Fine-grained text to image generation with attentional generative adversarial networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 1316-1324.
12. Zhu M, Pan P, Chen W, et al. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 5802-5810.
13. Tao M, Tang H, Wu F, et al. Df-gan: A simple and effective baseline for text-to-image synthesis[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 16515-16525.
14. Zhang L, Rao A, Agrawala M. Adding conditional control to text-to-image diffusion models[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2023: 3836-3847.
15. Ramesh A, Pavlov M, Goh G, et al. Zero-shot text-to-image generation[C]//International conference on machine learning. Pmlr, 2021: 8821-8831.
16. Ramesh A, Dhariwal P, Nichol A, et al. Hierarchical text-conditional image generation with clip latents[J]. arXiv preprint arXiv:2204.06125, 2022, 1(2): 3.
17. Saharia C, Chan W, Saxena S, et al. Photorealistic text-to-image diffusion models with deep language understanding[J]. Advances in neural information processing systems, 2022, 35: 36479-36494