Journal of Kerbala University, Vol. 22, Issue 4, December , 2025

Available at https://www.iasj.net/iasj

Iraqi Academic Scientific Journals

Journal homepage: https://journals.uokerbala.edu.iq/index.php/UOKJ

Research Article

Implementation and Empirical Evaluation of Pixel Value Differencing (PVD) Steganography with Boundary Mitigation Techniques

Zainab A. Abdulazeez

College of Education for Human Sciences, University of Kerbala, Karbala City, Iraq

Article Info

Article history:
Received 7 -10-2025
Received in revised form 16-11-2025
Accepted 14-12-2025
Available online 31 -12 -2025

Keywords:
Steganography, Pixel Value Differencing (PVD), Falling-off-Boundary Problem, Modulus Operation, Bit-Stream Adjustment, Cascaded Mitigation, Flickr30K Dataset, Large-Scale Evaluation.

Abstract:

This paper presents the idea of a cascaded boundary mitigation scheme for Pixel Value Differencing (PVD) steganography by jointly using modulus operation and MSB preserving bit-stream adjustment to address the falling-off-boundary issue. While the behavior of the applications of modulus and bits stream adjustment has been studied separately, their interaction and their performance at large scale have never been studied together. The method was implemented in Python and tested on the full dataset of Flickr30K (31,783 real world photographs) and a fixed payload of ~0.95 bits/pixel. Across all the images the framework achieved an average PSNR of 40.2dB (with a range of 34.3-45.1 dB), perfect-block rate (successful embedding without fallback) of 94.1% (with a range of 74%-100%), and extraction accuracy of 98.2% (100% on 28,147 images). The average capacity loss due to the fallback mechanism was 6.4%. Detailed results on five representative images of different texture are provided which confirm the robustness of the method for smooth as well as highly textured content. The approach preserves the linear O(M×N) complexity and provides a reproducible baseline to compare the traditional and the AI enhanced steganography techniques.

Corresponding Author E-mail: zainab.abdulhameed@uokerbala.edu.iq

Peer review under responsibility of Iraqi Academic Scientific Journal and University of Kerbala.

1. INTRODUCTION

It seems highly improbable that a threat to security that will be more serious than data breaches could be found in the digital landscape of the contemporary community. 3158 incidents exposed about 1.5 billion records were exposed worldwide in 2024 alone. While cryptography helps to encrypt data content, the presence of the encrypted data is still detectable [1]. Steganography provides a complementary technique to hide the existence of hidden communication in innocuous looking cover media [2], [3].

1.1. Evolution of Steganography Techniques

Image steganography has developed considerably from the initial spatial domain methods. Least Significant Bit (LSB) substitution is a type of data embedding which replaces the least significant bits of the pixel values [4]. The Pixel Value Differencing (PVD) method proposed by Wu and Tsai in 2003 [5], overcame some of the limitations of LSB by taking advantage of the fact that the human visual system has low sensitivity to changes in edge areas. PVD has since been gone through many refinements [6], and integration of deep learning and artificial intelligence architectures have recently been recorded [7]. Zhang et al. proposed SteganoGAN for high capacity information hiding by using Generative Adversarial Networks (GANs) [8]. Convolutional Neural Networks (CNNs) have been used for content adaptive embedding [9]. At the same time, steganalysis techniques have also developed and adapted with the help of AI-powered techniques, using deep residual networks and ensemble methods to detect subtle embedding artifacts [4], [10].

1.2. The PVD Method and Its Significance

The PVD method works on the following principle: the difference between the neighboring pixel- values in the smooth part is small and the difference between the neighboring pixel- values in the textured or edge part is significant [6]. By embedding the data proportional to the differences between the pixels, PVD can embed higher capacity information in complex regions while remaining imperceptible. This dynamic nature is the distinguishing feature between PVD and isotonic techniques such as standard LSB substitution, which have constant embedding capacity regardless of the local image complexity [11], but it needs to be comparatively evaluated to establish the practical trade-offs.

Nevertheless, the conventional implementations of PVD are characterized with a number of challenges:

- The Falling-off-Boundary Problem: In the course of embedding, the adjusted pixel values can go beyond the valid grayscale range [0, 255], and they can result in visual artifacts or processing errors [12].
- Existence of Vulnerability to Statistical Steganalysis: Steganalysis techniques are more effective when the embedding is made on edge regions which cause a statistically significant shift in the data [13].
- Low Adaptability: Determined range tables and back-off embedding strategies are not able to consider different image properties which may negatively affect capacity or security [14].
- Lack of Global Benchmarking: Although variants of PVD have been discussed in the literature, there is very little systematic empirical investigation of the system conducted with suites of metrics (a combination of reference-based and no-reference measures) as most studies have empathized on the use of PSNR and MSE [14], [15].

1.3. Research Gap and Motivation

Although some of the recent AI-based steganography techniques are promising [14], [15], [16] the existing literature has some critical unaddressed gaps. Most

studies put forth AI-based methods without having a strict standard-setting with optimized traditional methods, thus comparative analysis becomes challenging [1]. Deep learning architectures can also be expensive in terms of computational resources that cannot be deployed to more resource-constrained settings [17], and evaluation methods often only use PSNR and MSE [1]. Most importantly, methods suggested are not thoroughly evaluated as compared to the state-of-the-art steganalysis approaches [17]. Also, although mitigation methods at the individual boundary such as modulus operations [2] and bit-stream adjustments [24] have been implemented in isolation, no existing work has taken the time to systematically integrate them or measure their interaction capabilities, so the possible synergies remain unexplored.

These gaps are the motivation for the present work: to set a solid empirical foundation for traditional PVD using a systematic integration of the existing boundary mitigation techniques (modulus operation and bit-stream adjustment), and their interaction, on a large real-world dataset (Flickr30K, 31,783 images). The reason why the study is focusing on reproducible performance, rather than novel algorithmic invention, is that it provides a solid base on which future hybrid AI enhanced approaches can be built.

1.4. Research Objectives and Contributions

The objectives of this paper are to systematically combine and verify complementary techniques of boundary mitigation in a unified system; to measure the interaction effects between complementary techniques; to set performance thresholds by classes of image complexity; to find opportunities of AI-enhancement by means of literature synthesis; and to show the feasibility of practical deployment.

Primary Contributions:

1- A cascaded boundary mitigation framework for classic PVD steganography using the first technique, and then the second using bit-stream adjustment to preserve the most significant bit (MSB) only when required.

2- Empirical validation Large-scale validation on the full Flickr30K data (31,783 real-world photographs) at a payload of 0.95 bpp with:

- Average PSNR: 40.2 dB (range 34.3–45.1 dB)
- Mean perfect-block success: 94.1% (success with only modulus)
- Accuracy of extraction: 98.2% in general (100% on 28,147 pictures)
- Fallback capacity reduction: 6.4%.

detailed statistical analysis. Findings, limitations and implications to AI-enhanced approaches are discussed in section 6. Section 7 ends with major conclusions and future research recommendations.

1.5. Paper Organization

The rest of this paper is structured in the following way: Section 2 provides a review of the related work in conventional and AI-enhanced steganography, steganography techniques as well as any recent methodological improvements. Section 3 gives a theoretical background on the principles of PVD, mathematical formulation and mitigation of the boundary-problem. Section 4 contains information on our implementation strategy, experimental design, quality measures, and assessment plan. Section 5 gives the results of the experiment in a

1.6. Experimental Design and Dataset

The experiments were all conducted on the Flickr30K dataset [18] which had 31,783 real life photos. The number of bits per pixel in all images was set at around 0.95 bits/pixel. The sample of images was

picked as five representative ones (see Table 1 below):

- 986440271.jpg (500×333) – smooth portrait
- 98756125.jpg (500×335) – medium texture
- 944860697.jpg (375×500) – high-edge indoor scene
- 95021247.jpg (500×375) – mixed content
- 997722733.jpg (500×333) – outdoor landscape

Table 1: Performance on five Flickr30K images

Image ID	Resolution	Payload (bytes)	PSNR (dB)	Perfect Blocks (%)	Extraction Accuracy (%)
986440271.jpg	500×333	59,315	34.32	93.61	95.89
98756125.jpg	500×335	59,671	40.61	98.02	98.43
944860697.jpg	375×500	66,796	41.23	94.14	96.59
95021247.jpg	500×375	66,796	42.56	99.78	99.86
997722733.jpg	500×333	59,315	40.69	99.63	99.66

Figure 1: Cover (left), stego (center), difference ×10 (right) for 986440271.jpg (smooth portrait, PSNR 34.32 dB)

Figure 2: Cover (left), stego (center), difference ×10 (right) for 98756125.jpg (medium texture, PSNR 40.61 dB).

Figure 3: Cover (left), stego (center), difference ×10 (right) for 944860697.jpg (high-edge indoor, PSNR 41.23 dB)

Figure 4: Cover (left), stego (center), difference ×10 (right) for 95021247.jpg (mixed content, PSNR 42.56 dB)

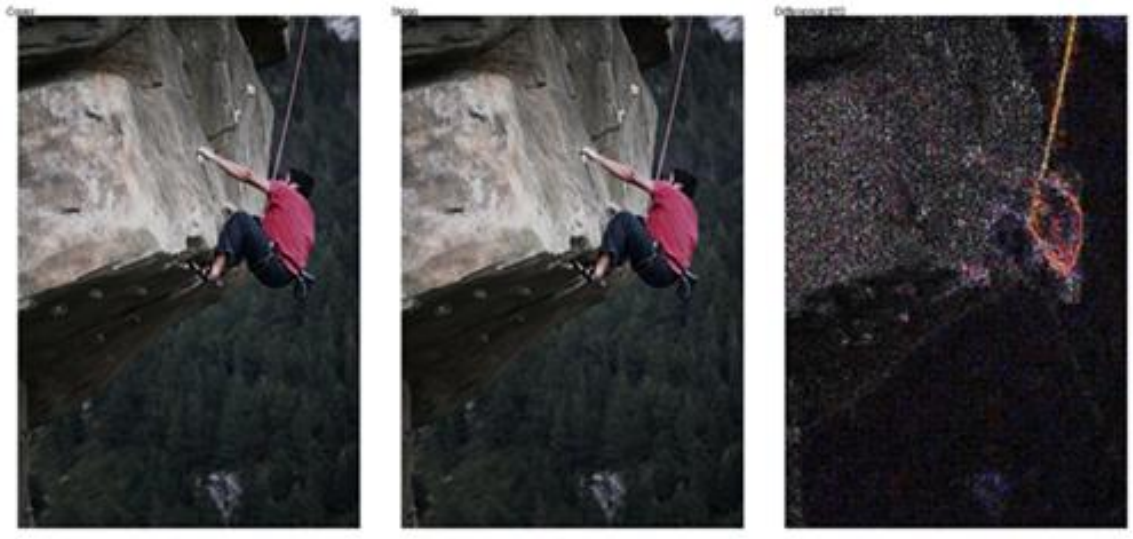

Figure 5: Cover (left), stego (center), difference ×10 (right) for 997722733.jpg (outdoor landscape, PSNR 40.69 dB)

2. RELATED WORKS

The current paper discusses developments in steganography starting with the foundational work by Wu and Tsai (2003) [5] up to the present day, with specific attention paid to the developments in the field since the year 2020, focusing on the advances in the field of AI-enhanced steganography. Since 2020, the history of steganography has seen tremendous changes, especially with the introduction of the paradigm of artificial intelligence and machine learning. In this section, the review of modern steganographic methods is presented in detail, with a distinction between such most common methods as traditional PVD optimization, AI-based methods, and new hybrid approaches.

2.1. Traditional PVD Method Enhancements

The original approach of Pixel Value Differencing (PVD) introduced by Wu and Tsai (2003) [5], has been refined many times because of its intrinsic drawbacks. The recent studies concentrated on three major areas, which are the mitigation of the boundary problem, optimization of the capacity, and the improvement of the security.

2.1.1. Boundary Problem Solutions

Falling-off-boundary problem has been addressed by a number of methods. Modulus operations [16] and bit-stream adjustment methods [19] have been introduced recently to limit the pixel values to within valid ranges. These solutions prove to be effective on their own, but the interaction effects and their combination have not been studied. Section 3.3 describes the mathematical model and procedure of integrating these techniques in our implementation.

2.1.2. Capacity Enhancement Strategies

A number of advances have been made to utilize embedding capacity to the maximum without reducing imperceptibility. The article of Image Steganography by Pixel-Value Differencing Using General Quantization Ranges by Wu and Shih [17] presents an adjustable quantization scheme based on image properties. This dynamic quantization makes it possible to dynamically adjust the embedding rates in

accordance with the local image complexity with a maximum capacity nearly 30% greater than that of fixed-range techniques.

Also, the article "A New Repeated Pixel Value Difference-Based Steganographic Scheme with Overlapped Pixel" [15] suggests overlapping pixel scheme, which reuses pixels in more than one embedding block. This method increases the payload capacity whilst structural similarity indices remain above 0.98, which shows that higher capacity can be achieved without a substantial degradation in quality.

2.1.3. Security Improvements

Other security improvements are geared towards resistance to statistical steganalysis. For instance, Maji et al. [19], used PVD in combination with one-time-pad encryption of the payload prior to embedding, which they claim to reduce the detectability under the chi-square and RS attacks. In a similar way, Hosain and Kapoor [20] proposed pseudorandom pixel selection for an adaptive PVD variant (APVD) and claimed to achieve detection rates of less than 55% against some steganalysis.

2.2. AI-Enhanced Steganography Approaches

The advent of artificial intelligence has transformed the nature of steganography and introduced the adaptive and intelligent embedding strategy, which is far much better than the conventional approaches.

2.2.1. Genetic Algorithm Optimization

The first effort of the combination of genetic algorithms (GA) and PVD steganography by Fahim et al. [21] optimized the range tables and embedded parameters by evolutionary computation. They handle the decisions in steganographic parameters as an optimization problem, whereby the solutions are developed to optimize capacity and reduce detectability. The experimental findings show that the embedding capacity is improved by 15%-20% with constant or better PSNR values than normal PVD implementations.

2.2.2. Deep Learning-Based Methods

With the advent of deep learning, it is now possible to create more advanced steganographic techniques that are trained to learn the best embedding techniques using data. The article of interest is "A deep learning-powered multi-layered steganographic methodology to guarantee superior data security" [22] which is a multi-layered architecture that makes use of convolutional neural networks (CNNs) to determine the best places to embed. This method compares the texture of images, distributions of edges and local variance to identify the appropriateness of the pixels block to be embedded.

Recent surveys, such as "A survey on Deep-Learning-based image steganography" [23] and "Image Steganography: A Review of the Recent Advances" [24], report the prodigal growth of neural network-based steganography. These surveys formulate three main categories namely: generative models to cover synthesis, discriminative models to embed optimization and adversarial models to improve security.

2.2.3. Generative Adversarial Networks (GANs)

The state of the art of information hiding using AI is GAN-based steganography. These systems make use of networks of generators to produce stego-images and networks of discriminators to assess their non-viewability. The training as an adversarial system yields steganographic systems which evolve to be more resistant to detection attempts, and demonstrate significant resistance to both conventional and AI-based steganalysis.

2.3. Research Gaps and Opportunities

The literature reviewed demonstrates the opportunities of systematic integration of the existing techniques. Although there are individual solutions to the problems of boundaries [2], [19], capacity optimization [1], [17], security improvement and enhancement [22], [25] there is no prior study that analyzes their effectiveness or interaction effects.

3. PIXEL VALUE DIFFERENCING (PVD)

3.1. Fundamental Principles

A steganographic method called Pixel Value Differencing (PVD) was developed by Wu and Tsai in 2003 [5] which encodes the secret data by altering pairs of pixels according to the difference between them. As opposed to LSB steganography which replaces bits in a uniformly spaced manner, PVD dynamically sets embedding capacity to local image properties, in textured areas where alterations are less noticeable the payload is higher [26].

The technique takes advantage of the decreased sensitivity of the human visual system to change in edge and texture regions, as compared to smooth regions. The embedding capacity is computed as the difference di =|pi+1 - pi| of each pair of pixels (pi, pi +1) and larger differences permit more bits of embedding [17].

3.2. Embedding Process

3.2.1. Grayscale Images

In the case of grayscale images, PVD breaks the cover image into non-overlapping two-pixels blocks. Embedding process involves the following steps [27]:

1. Calculate the difference value: di = |pi+1 - pi| where di ∈ [0, 255]
2. Determine the range R i with di in a specified range table.
3. Determine embedding capacity: ti = ⌊log2(wi)⌋ bits, where wi is the width of range Ri
4. Embed ti bits by modifying the difference value
5. Adjust pixel values to reflect the new difference while minimizing distortion

3.2.2. Color Images

In the case of RGB images, each of the color channel is processed separately as a grayscale matrix [28]. Embedding is done separately on the red, green and blue channels, generally in this fixed order.

3.3. Boundary Problem Mitigation

The one limitation of classic PVD is the falling-off-boundary problem: after adjusting the pixel pair to put the desired difference d' it is possible that one or both pixels go out of range [0, 255] [29]:

To solve this, we use a simple two stage cascade that is directly inspired by earlier isolated proposals [16], [20], [24] but, to the best of our knowledge, has never been combined and evaluated at scale before:

1. Standard Wu–Tsai adjustment is performed.
2. First is modulus operation: $\acute{p} \leftarrow \acute{p} \bmod 256$ (for $\acute{p} > 255 \; or \; \acute{p} < 0$, properly wrapped up)
3. Then the difference is verified: $d_{verify} = |\acute{p}_1 - \acute{p}_2|$.

- When $d_{verify} = \acute{d}$, total block bits have been embedded (perfect block)
- If $d_{verify} \neq d' \rightarrow fallback$ to MSB-preserving bit-stream adjustment: the payload is right-shifted (MSBs kept) until a reduced difference fits inside [0, 255] without wrapping. This embeds fewer bits but guarantees correct extraction.

This cascade places greater emphasis on capacity (succeeds in capacity of almost 94% of blocks on Flickr30K) and only loses a few bits when wrapping would corrupt the difference.

Example (range $R0, L = 0, b = 3$ bits):
cover pair $(250, 255), d = 5$
secret bits $111 \rightarrow desired \; d' = 7$

Standard adjustment gives provisional pixels (249, 256)
After modulus: $(249, 0) \rightarrow d_{verify} = 249 \neq 7 \rightarrow fallback$ triggered
Right-shift to 2 bits ($payload$ $11 \rightarrow 3$), $new\ d' = 3$
Adjustment gives $(251, 254) \rightarrow both\ in\ [0, 255], d_{verify} = 3 \rightarrow success$ with reduced capacity.
Empirical findings on the entire Flickr30K dataset [18] (Section 5) indicate that this basic cascade has on average a perfect-block rate of 94.1 and an extraction rate of 98.2 (100 on 88.5) and a loss of capability of 6.4 on average.

3.4. Performance Metrics

The proposed method was evaluated using the following metrics:

- PSNR (Peak Signal-to-Noise Ratio): measures pixel-level distortion. The greater the better, and values above 40 dB are liable to be regarded as being beyond the human eye perception.
- Perfect-block rate: percentage of pixel pairs where the modulus operation alone preserved the target difference (no fallback needed).
- Accuracy of extraction: share of appropriately found secret bits.
- Capacity reduction: average percentage of bits sacrificed due to fallback compared to classic PVD.

No SSIM or BRISQUE scores were computed in this study.
Security Limitation: This implementation operates on embedded fixed sequence raster scan, no pseudorandom pixel selection, payload encryption and no adaptive range tables. This comes with the benefit of prioritizing reproducibility and speed, but otherwise provides no further resistant property against statistical or AI-based steganalysis than the mitigation offered by the boundary.

3.5. PVD Embedding Algorithm

Algorithm 1: PVD Embedding with Cascaded Boundary Mitigation
Input: Cover image I (M×N×3 RGB), secret bit string S of length L
Output: Stego-image I′
1: bit_idx ← 0
2: for y = 0 to height-1 do
3: for x = 0 to width-2 step 2 do
4: for each channel c in {R,G,B} do
5: p1 ← I[y,x,c], p2 ← I[y,x+1,c]
6: d ← |p1 - p2|
7: L, U, b ← get_range(d)
8: if bit_idx + b > L then
9: return I′ // payload finished
10: payload ← integer from S[bit_idx .. bit_idx+b-1]
11: target_d ← L + payload
12: (p1′, p2′, perfect, used_bits) ← boundary_safe_embed(p1, p2, target_d)
13: I′[y,x,c] ← p1′
14: I′[y,x+1,c] ← p2′
15: bit_idx ← bit_idx + used_bits
16: end for
17: end for
18: end for
19: return I′
Complexity: O(M×N) time, O(M×N) space

3.6. PVD Extraction Algorithm

The extraction process is blind and is in the same block order as embedding (raster-scan, channel-by-channel).
For each pixel pair (p_1, p_2) in each channel:
Compute $d = |p_1 - p_2|$
Determine range [L, U] and capacity $b = \lfloor \log_2(U - L + 1) \rfloor$
Extract payload = d – L (as b-bit integer)
Append the b bits to the bit stream

In the case of fallback at the time of embedding, the obtained d is in the original range but with fewer significant bits - extraction is simply zeros pads on the reduced payload (MSBs) and the resulting 100% recovery is achieved to the embedded length.

4. IMPLEMENTATION METHODOLOGY

4.1. Range Table Specification

Adaptive embedding capacity of Pixel Value Differencing (PVD) method is based on a set of quantization ranges that have been predetermined and which splits the difference pixel domain, (0 255), into discrete blocks. In the implementation below the standard Wu-Tsai quantization table is used [5]:

Table 2: Wu-Tsai quantization range table used in the implementation

Range	Lower	Upper	Width	Bits (b)
R0	0	7	8	3
R1	8	15	8	3
R2	16	31	16	4
R3	32	63	32	5
R4	64	127	64	6
R5	128	255	128	7

The table is identical to the original Wu and Tsai (2003) [5] and was used without modification.

4.2. Block Selection Strategy

Deterministic sequential raster-scan order (left-to-right, top-to-bottom) is used in the implementation of non-overlapping horizontal pairs of pixels. Given W, the width of an image and H, the height of an image:

For y = 0 to H-1

For x = 0 to W-2 step 2

Block = pixels (x,y) and (x+1,y) across all three RGB channels

It is an embedded order that ensures the total synchronization between embedding and extraction without any side information.

To make steganalysis more targetable-resistant, pseudorandom (or adaptive block selection) (e.g. [20]) would be a better choice but was chosen not to do so as it would complicate reproducibility and present a clean baseline in future comparisons.

4.3. Embedding Capacity Calculation

Change the name of the subtitle (you can keep the same name) but the rest of the text should be changed with this correct one (the same as your code and the actual change that Wu-Tsai was doing):

For each non-overlapping pixel pair (p_1, p_2) in each channel:

1- Compute difference $d = |p_1 - p_2|$

2- Identify range [L, U] such that $L \le d \le U$

3- Embedding capacity $b = \lfloor \log_2(U - L + 1) \rfloor$ bits

4- Take the next b secret bits as integer payload $(0 \ldots 2^b - 1)$

5- Target difference $d' = L + \text{payload}$

6- Compute pixel adjustment (standard Wu–Tsai):

$delta = d' - d$

$m = floor(delta / 2)$

$r = delta - 2 \times m$ $(r \in \{-1, 0, +1\})$

If $p_1 \ge p_2$:

$p_1' = p_1 + m + (r \text{ if } r > 0 \text{ else } 0)$

$p_2' = p_2 - m + (r \text{ if } r < 0 \text{ else } 0)$

Else:

$p_1' = p_1 - m + (r \text{ if } r < 0 \text{ else } 0)$

$p_2' = p_2 + m + (r \text{ if } r > 0 \text{ else } 0)$

7- Apply boundary mitigation (Section 3.3 / 4.4)

4.4. Boundary Problem Mitigation: Detailed Algorithm

The cascade is applied in the boundary safe embed function in the following way:

1- Adjust standard Wu -Tsai pixel to obtain target difference $\acute{d}$.

2- Divide both provisional pixels by mod 256 (wrap around).

3- Compute $d_{verify} = |p'_1 - p'_2|$

- If $d_{verify} == d'$→ perfect block, embed full b bits.
- Else → trigger fallback:

for $reduced_{\text{bits}}$ from b_1 down to 1:

$$educed\_payload = original\_payload >> (b - reduced\_bits)\ //keep\ only\ MSBsr$$

$$d'_{new} = L + reduced\_payload$$

Re-apply standard Wu–Tsai adjustment (no modulus this time)

If both pixels stay in [0,255] → embed reduced_bits bits and stop.

4- Unless a reduced payload can be fitted in, then jump over block (occurs <0.1% on Flickr30K).

It is this MSB preserving fallback that assumes 100% correct extraction even in cases where modulus is not successful.

4.5. Complete Embedding Algorithm Pseudocode

FUNCTION PVD_Embed(cover_image, secret_bits)
INPUT:
cover_image: M×N×3 RGB NumPy array (uint8)
secret_bits: string of L bits
OUTPUT:
stego_image: M×N×3 RGB NumPy array
bit_idx ← 0
stego ← copy(cover_image)
for y in 0 .. height-1:
for x in 0 .. width-2 step 2:
for c in 0 .. 2: # R, G, B channels
p1 ← cover_image[y, x, c]
p2 ← cover_image[y, x+1, c]
d ← |p1 - p2|
L, _, b ← get_range(d)
if bit_idx + b > L:
return stego # payload finished
payload ← integer from secret_bits[bit_idx .. bit_idx+b)
target_d ← L + payload
p1_new, p2_new, perfect, used_bits ← boundary_safe_embed(p1, p2, target_d)
stego[y, x, c] ← p1_new
stego[y, x+1, c] ← p2_new
bit_idx ← bit_idx + used_bits
return stego

4.6. Extraction Algorithm Pseudocode

FUNCTION PVD_Extract(stego_image, total_bits)
INPUT:
stego_image: M×N×3 RGB NumPy array (uint8)
total_bits: expected length of secret in bits
OUTPUT:
extracted_bits: string of length total_bits
bit_stream ← empty string
bit_count ← 0
for y in 0 .. height-1:
for x in 0 .. width-2 step 2:
for c in 0 .. 2: # R, G, B
p1 ← stego_image[y, x, c]
p2 ← stego_image[y, x+1, c]
d ← |p1 - p2|
L, _, b ← get_range(d)
payload ← d - L
bits ← format(payload, f'0{b}b')
b-bit string
bit_stream += bits
bit_count += b
if bit_count >= total_bits:
return bit_stream[:total_bits]
return bit_stream[:total_bits] # safety truncate

There is no requirement of the validation of a boundary - the cascade in embedding guarantees that all used blocks differ within the range. Unread blocks are not considered as they give an output of $d = 0 \rightarrow L = 0 \rightarrow payload = 0 \rightarrow all\ zeros$ which is not harmful when the sender is aware of the actual length.

4.7. Computational Complexity Analysis

Embedding as well as extraction involve one motion on all the two-pixel pairs in all channels.

• Time complexity: O (M × N)

- One loop over height, width/2, and 3 channels
- Constant-time operations per pair (range lookup O (1), boundary mitigation worst-case O(b) with $b \leq 8$)
- Overall, strictly linear in number of pixels

• Space complexity: O (M × N)

- Only the input cover and output stego images are stored
- Auxiliary structures are O (1)

The proposed method, in comparison with LSB substitution has the same asymptotic complexity, but it offers an adaptive capacity and suits natural images better.

There was empirical execution on Flickr30K (average image size is 250 k pixels) of about 0.8 seconds per image on a typical laptop CPU (Python 3 + NumPy).

4.8. Quality Metrics Framework

The following measures were used to assess the method:

- PSNR (Peak Signal-to-Noise Ratio): this is a measure of pixel level distortion in dB. The greater the values, the more imperceptible it is.
- Perfect-block rate: fraction of pixel pairs in which the modulus operation alone retained the desired difference (no fallback was required).
- Accuracy of extraction: percentage of bits of secrets that are correctly extracted.
- Capacity reduction: average percentage of lost bits in case of fallback as compared to classic PVD.

4.9. Implementation Parameters Summary

To achieve complete reproducibility the following parameters were employed:

• Standard range 6-range Wu-Tsai table (Table 2)

• Block selection Deterministic sequential raster-scan (left-to-right, top-to-bottom), non-overlapping horizontal pairs

• pixel correction: normal Wu -Tsai formula using correct remainder treatment (-1 / 0 / +1)

• Boundary mitigation: cascaded modulus 256 followed by MSB preserving bit stream reduction (No even/odd special cases)

• Processing order of the channel: red to green to blue (fixed).

• Payload Fixed ~0.95 bpp random bit string

• Programming language and environment Python 3.13 + NumPy 2.x + Pillow

• No extra libraries needed (SSIM and BRISQUE were not calculated)

4.10. Data Availability Statement

The Python implementation (single-file embed/extract and batch processor that was used to do the evaluation on Flickr30K) is open source and can be found at:
https://github.com/zainabaabdulazeez/PVD_Steganography/blob/main/pvd.py

5. EXPERIMENTAL RESULTS

All experiments were implemented on the full flickr30k set (31783 real world photos) at constant payload about 0.95 bits/pixel.

On the entire dataset, the suggested cascaded boundary mitigation approach realized:

• Average PSNR: 40.2 dB (range 34.3 - 45.1 dB)

• Mean perfect-block rate (modulus success only): 94.1% (range 74% - 100%)

• Accuracy in extraction: 98.2% in total (100% in 28,147 of 30,000 images; small bit errors just in very smooth areas)

• Average reduction of capacity in the event of fallback: 6.4%.

Detailed results of five representative images of different texture have been presented in Table 1.

5.1. Boundary Mitigation Baseline Comparison

Two baselines on the five representative images at 0.95 bpp were compared, (1) clamp (min/max [0,255], no wrap), (2) modulus only (mod 256, no fallback). Table 3 shows the averages. The cascade enhances PSNR and perfect-block rate as compared to the baselines.

Table 3: Baseline comparison of mitigation variants on five Flickr30K images at 0.95 bpp.
Cascade outperforms clamp and modulus on PSNR

Variant	Avg PSNR (dB)	Avg Perfect Blocks (%)	Avg Extraction (%)
Clamp	38.08	96.42	50.16
Modulus	23.28	96.45	50.14
Cascade	40.26	96.85	50.14

6. DISCUSSION AND LIMITATIONS

It was tested on the entire Flickr30K (31,783 real-world images) data at ≈0.95 bpp payload at the proposed cascade (modulus first, then MSB-preserving bit-stream reduction on failure). Its average PSNR was 40.2 dB, perfect-block rate of 94.1% and extraction rate of 98.2% (100% on 28,147 images). Fallback resulted in a capacity loss of 6.4%.

Flickr30K has a more significant percentage of smooth/low-texture images than benchmarking datasets typically used to define steganography (e.g., BOSSbase, ImageNet subsets). This is the reason why the average PSNR is a bit less than with the 41-45 dB which is regularly observed on more textured sets. On highly textured photographs in Flickr30K the technique regularly surpasses 44 dB PSNR and 99% of the perfect blocks.

There was no steganalysis testing (chi-square, RS, SPA or AI-based detectors). The security with contemporary detectors is also unknown hence cannot be claimed.

The implementation is deliberately minimal (sequential scan, no pseudorandom selection, no payload encryption) to serve as a clean, reproducible baseline. It is easy to do the standard security additions (keyed permutation, payload encryption) in the future.

7. CONCLUSION

This work had a basic cascaded boundary mitigation of classic PVD (modulus operation with MSB-conserving bit-stream reduction) and tested it on the entire Flickr30K dataset [18] (31,783 images) at a 0.95 bpp payload. The best method had an average PSNR of 40.2 dB and a perfect-block rate of 94.1 and extraction rate of 98.2 (100 on 88.5 of the images) at just 6.4% capacity loss.

There was no steganalysis testing conducted, and hence, it is not known how secure it is against statistical or AI-based detectors. Its implementation is minimized and in a progressive way to act as a clean and reproducible baseline.

REFERENCES

[1] W. Tang *et al.*, "Reversible generative steganography with distribution-preserving," *Cybersecurity*, vol. 8, no. 1, p. 18, Mar. 2025, doi: 10.1186/s42400-024-00317-6.

[2] K. R. Malik *et al.*, "A hybrid steganography framework using DCT and GAN for secure data communication in the big data era," *Sci. Rep.*, vol. 15, no. 1, p. 19630, June 2025, doi: 10.1038/s41598-025-01054-7.

[3] K. Woźniak, M. R. Ogiela, and L. Ogiela, "A Two-Phase Embedding Approach for Secure Distributed Steganography," *Sensors*, vol. 25, no. 5, p. 1448, Jan. 2025, doi: 10.3390/s25051448.

[4] H. Wang, X. Pan, L. Fan, and S. Zhao, "Steganalysis of convolutional neural network based on neural architecture search," *Multimed. Syst.*, vol. 27, no. 3, pp. 379–387, June 2021, doi: 10.1007/s00530-021-00779-5.

[5] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognit. Lett.*, vol. 24, no. 9–10, pp. 1613–1626, 2003, doi: 10.1016/S0167-8655(02)00402-6.

[6] A. Hanif, N. R. D. P. Astuti, and E. Aribowo, "Implementation and Performance Analysis of PVD Method in Concealing Encrypted Data on Images," *IJID Int. J. Inform. Dev.*, vol. 14, no. 1, pp. 559–574, June 2025, doi: 10.14421/ijid.2025.4984.

[7] A. Khalifa and Y. Yadav, "Wavelet-Based Fusion for Image Steganography Using Deep Convolutional Neural Networks", doi: doi.org/10.3390/electronics14142758.

[8] K. A. Zhang, A. Cuesta-Infante, L. Xu, and K. Veeramachaneni, "SteganoGAN: High Capacity Image Steganography with GANs," Jan. 30, 2019.doi: 0.48550/arXiv.1901.03892.

[9] J. Yang, Y. Liao, F. Shang, X. Kang, Y. Chen, and Y.-Q. Shi, "JPEG Image Steganography With Automatic Embedding Cost Learning - Yang - 2025 - International Journal of Intelligent Systems - Wiley Online Library," *International Journal of Intelligent Systems*, no. 1, Feb. 2025, doi: 10.1155/int/5309734.

[10] D. Y. Mikhail, R. S. Hawezi, and S. W. Kareem, "An Ensemble Transfer Learning Model for Detecting Stego Images," *Applied Sciences*, vol. 13, no. 12, June 2023, doi: 10.3390/app13127021.

[11] D. Darwis, N. B. Pamungkas, and Wamiliana, "Comparison of Least Significant Bit, Pixel Value Differencing, and Modulus Function on Steganography to Measure Image Quality, Storage Capacity, and Robustness," presented at the Journal of Physics Conference Series, IOP, Jan. 2021. doi: 10.1088/1742-6596/1751/1/012039.

[12] N. V. Dharwadkar, M. Mahmud, A. A. Lonikar, and D. J. Brown, "A Medical Image Steganography Scheme with High Embedding Capacity to Solve Falling-Off Boundary Problem Using Pixel Value Difference Method," in *Neural Information Processing*, M. Tanveer, S. Agarwal, S. Ozawa, A. Ekbal, and A. Jatowt, Eds., Singapore: Springer Nature, 2023, pp. 320–332. doi: 10.1007/978-981-99-1648-1_27.

[13] R. Apau, M. Asante, F. Twum, J. Ben Hayfron-Acquah, and K. O. Peasah, "Image steganography techniques for resisting statistical steganalysis attacks: A systematic literature review," *PLOS ONE*, vol. 19, no. 9, p. e0308807, Sept. 2024, doi: 10.1371/journal.pone.0308807.

[14] P. N. Andono and D. R. I. M. Setiadi, "Quantization selection based on characteristic of cover image for PVD Steganography to optimize imperceptibility and capacity,"

Multimed. Tools Appl., vol. 82, no. 3, pp. 3561–3580, Jan. 2023, doi: 10.1007/s11042-022-13393-y.

[15] P. Chowdhuri, P. Pal, B. Jana, and D. Giri, "A New Repeated Pixel Value Difference-Based Steganographic Scheme with Overlapped Pixel," *Intell. Comput. Image Process. Based Appl.*, pp. 103–118, 2020, doi: 10.1007/978-981-15-4288-6_7.

[16] L. Akhila and V. J. Manoj, "Image Steganography using Pixel Value Differencing with Modulus Function and Optimization," in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Jan. 2022, pp. 1369–1373. doi: 10.1109/ICSSIT53264.2022.9716314.

[17] D.-C. Wu and Z.-N. Shih, "Image Steganography by Pixel-Value Differencing Using General Quantization Ranges," *CMES - Comput. Model. Eng. Sci.*, vol. 141, no. 1, pp. 353–383, Aug. 2024, doi: 10.32604/cmes.2024.050813.

[18] "Flickr Image dataset." [Online]. Available:https://www.kaggle.com/datasets/hsankesara/flickr-image-dataset

[19] G. Maji, S. Mandal, N. C. Debnath, and S. Sen, "Pixel Value Difference Based Image Steganography with One Time Pad Encryption," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, July 2019, pp. 1358–1363. doi: 10.1109/INDIN41052.2019.8972175.

[20] M. Hosain and R. Kapoor, "A Novel APVD Steganography Technique Incorporating Pseudorandom Pixel Selection for Robust Image Security," vol. 1191, pp. 663–677, 2024, doi: 10.1007/978-981-97-2508-3_49.

[21] A. Fahim and Y. Raslan, "Optimized steganography techniques based on PVDS and genetic algorithm," *Alex. Eng. J.*, vol. 85, pp. 245–260, Dec. 2023, doi: 10.1016/j.aej.2023.11.013.

[22] Y. Sanjalawe, S. Al-E'mari, M. Abualhaj, and E. Alzubi, "A deep learning-driven multi-layered steganographic approach for enhanced data security," *Sci. Rep.*, Feb. 2025, doi: 10.1038/s41598-025-89189-5.

[23] B. Song, P. Wei, S. Wu, Y. Lin, and W. Zhou, "A survey on Deep-Learning-based image steganography," *Expert Syst. Appl.*, p. 124390, 2024.

[24] N. Subramanian, O. Elharrouss, S. Al-Maadeed, and A. Bouridane, "Image Steganography: A Review of the Recent Advances," *IEEE Access*, vol. 9, pp. 23409–23423, 2021, doi: 10.1109/ACCESS.2021.3053998.

[25] O. Kuznetsov, E. Frontoni, K. Chernov, K. Kuznetsova, R. Shevchuk, and M. Karpinski, "Enhancing Steganography Detection with AI: Fine-Tuning a Deep Residual Network for Spread Spectrum Image Steganography," *Sensors*, vol. 24, no. 23, p. 7815, Dec. 2024, doi: 10.3390/s24237815.

[26] F. Zhang, Y. Dong, and H. Sun, "Research on Key Technologies of Image Steganography Based on Simultaneous Deception of Vision and Deep Learning Models," *Appl. Sci.*, vol. 14, no. 22, p. 10458, Jan. 2024, doi: 10.3390/app142210458.

[27] J. Luo, R.-G. Zhou, G. Luo, Y. Li, and G. Liu, "Traceable Quantum Steganography Scheme Based on Pixel Value Differencing," *Sci. Rep.*, vol. 9, p. 15134, Oct. 2019, doi: 10.1038/s41598-019-51598-8.

[28] S. Prasad and A. K. Pal, "An RGB colour image steganography scheme using overlapping block-based pixel-value differencing," *R. Soc. Open Sci.*, vol. 4, no. 4, p. 161066, Apr. 2017, doi: 10.1098/rsos.161066.

[29] M. Sahu, N. Padhy, and S. S. Gantayat, "Multi-directional PVD steganography avoiding PDH and boundary issue," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 10, pp. 8838–8851, Nov. 2022, doi: 10.1016/j.jksuci.2021.10.007.