

# Pawprint v2024b: A Production-Grade Traditional Image Watermark

Bill Huang, Jiaxin Guan, and Cindy Han  
Trufo Inc.

## Abstract

As the quality and quantity of content generated by AI continues to increase, the demand for watermarking solutions—either to identify synthetic content or to verify camera-captured and human-created content—has grown substantially over the past few years. This submission presents Pawprint v2024b, a production-grade watermark developed by Trufo Inc. in 2024, entered into the JPEG Trust Watermarking Benchmark ICIP 2026 Grand Challenge. This particular version of Pawprint does not make use of neural network techniques, and instead relies entirely on traditional methods, achieving an average BER of under 2% at 100 bits and an average peak signal-to-noise ratio (PSNR) of 47 on the competition evaluation.

## 1 Introduction

Image watermarking became popular in the 1990s, especially for discouraging piracy and protecting copyright. Early work such as secure spread-spectrum watermarking showed that multimedia content could carry hidden signals while preserving perceptual quality [1]. Since then, watermarking has been studied across a variety of domains, starting with spatial-domain methods before evolving into transform-domain methods based on the discrete Fourier, cosine, and wavelet transforms (DFT, DCT, and DWT).

In 2018, deep learning techniques were introduced for image watermarking with HiDDeN, which trained neural encoders and decoders to hide binary messages in images [2]. Deep learning has since advanced the field substantially, and most state-of-the-art image watermarks are now based on neural networks, including PixelSeal, SynthID, and TrustMark [4–6]. In contrast to the neural network trend, this paper presents a traditional image watermark, Pawprint v2024b that is able to keep up with the advances made so far.

Historically, image watermarks were often designed for copyright protection, where oftentimes extraction is semi-blind: the decoder may use side information from the original image or the embedding process. This is substantially easier than blind decoding, and many traditional watermarks, such as DWT-DCT-SIFT [3], attain durability and perceptibility scores in the non-blind setting similar to what Pawprint v2024b attains in the blind setting.

More recently, watermarking has also become useful for content provenance, where a watermark can identify content after publication and distribution so that provenance information can be fetched from the cloud. In this setting, blind extraction is important because the decoder does not have the original content on hand. For example, Trufo uses watermarks to embed an ID into an image, and this ID is used to fetch data, such as a Coalition for Content Provenance and Authenticity (C2PA) manifest signed by the original publisher, from a secure cloud

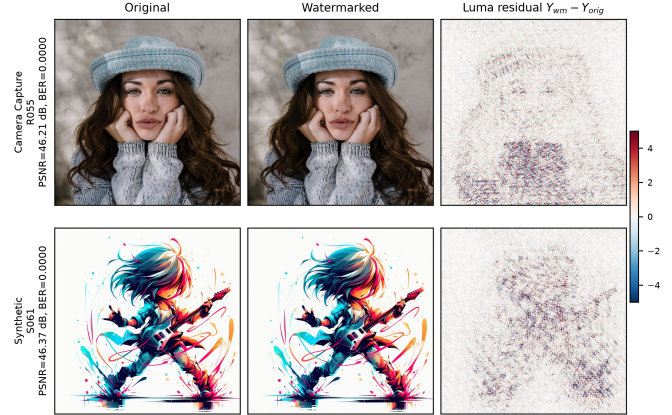


Figure 1: Representative Pawprint v2024b embedding examples using images R055 from the Camera Capture subset and S061 from the Synthetic subset of the JPEG Trust benchmark dataset. The residual panels show the luma difference  $Y_{wm} - Y_{orig}$ ; the watermarks are visually imperceptible while remaining machine-readable.

database [7].

Additionally, the watermark needs to be durable to common image edits, invisible—generally with PSNR above 45—and scalable, with fast encoding and decoding. For commercial adoption, a PSNR below 45 interferes with content quality, while slow performance and heavy dependencies interfere with client workflows. Durability against simple cropping, rotation, color filters, compression, alterations, etc. are necessary for the watermark to be useful in practice. Existing traditional watermarks fall short of these new requirements.

The security attack surface also shifts: because the watermark now carries positive value instead of negative value, resilience against adversarial removal attacks is less important, while resilience against forgery is more important. In Trufo’s ecosystem, the protection against forgery attacks includes components that live outside the watermark, such as cryptographic signatures and perceptual hashes.

Pawprint v2024b improves upon existing traditional watermarks by an order of magnitude, and still beats the SotA neural network watermarks of 2026 in many metrics. Pawprint v2024b is a modified version of Trufo’s initial 2024 image watermarking product, adjusted to carry 100 bits instead of the original 52 bits.

## 2 Method

The six main techniques that were introduced and developed for Trufo’s Pawprint v2024b watermark are described below, in brief detail:

### 2.1 Algorithm Details

(1) A signal separation transform  $s(I)$  is applied to the image. This transform is a convolution kernel designed to preserve fea-

tures that are relevant to the watermark, and reduces the dynamic range of the image from  $[0, 255]$  to  $(-8, +8)$ .

(2) The watermark pattern is embedded in DFT space, and is separated into an alignment pattern and a message payload pattern. The alignment pattern is used to correct for geometric transformations (cropping, rotation), by deriving a rotation matrix and then inverting it to realign the image.

(3) The pattern suppresses the magnitudes of certain features in the feature space: instead of adding a large signal to a set of frequencies, it substantially reduces the existing signal in a selected set of frequencies. This enables higher statistical confidence in decoding.

(4) Each message payload symbol conveys 5 bits of information: each value of each symbol corresponds to a set of features that are suppressed, so overall 30 sets of features are suppressed, out of a total of  $30 \times 32 = 960$  sets of features that could be suppressed. This enables us to make fewer changes to communicate the same number of bits.

(5) A Reed-Solomon scheme of  $[30, 20]$  over  $\text{GF}(2^5)$  is used for error correction, carrying the 100-bit payload; the alphabet size fits well with the 5-bit symbol design.

(6) An amplification shape  $a(I)$  is computed based on the image. This scales the watermark pattern based on the local activity in the spatial region, so that most of the watermark is embedded in areas where the watermark features are less noticeable.

Pawprint v2024b differs from the original production version (v2024) in that the payload is 100 bits ( $\text{RS}_{2^5}[30, 20]$ ) instead of 52 bits ( $\text{RS}_{2^4}[15, 13]$ ).

## 2.2 Implementation Details

The implementation is in Python, but makes use of FFT kernel libraries and numba JIT compilation to improve speed (though if pytorch is used for certain calculations, including the FFT, a 10x speedup can be achieved on the evaluation hardware).

## 3 Evaluation

We compare Pawprint v2024b (no-ML, 100-bit) against two popular watermarks: a modified version of Meta’s PixelSeal with our own error correction logic (PixelSeal+BCH), and a standard version of the popular invisible-watermark library’s DWT-DCT-SVD watermark. The evaluation is run on a NVIDIA 5090 GPU and an AMD 7950X3D CPU. The comparison is conducted with the JPEG Trust Challenge benchmark, consisting of 100 camera-captured images and 100 synthetic images. The benchmark includes signal-processing attacks, geometric attacks, compression attacks, and AI modification attacks. In addition to standard PSNR/SSIM, we developed a wPSNR to more accurately measure watermark visibility, by being strict in flat areas and lenient in noisy areas.

The results show that Pawprint v2024b substantially outperforms the DWT-DCT-SVD reference, in invisibility, in durability, and in payload size. PixelSeal+BCH edges out Pawprint v2024b in a small majority of categories in this benchmark, while Pawprint v2024b edges out PixelSeal+BCH slightly in PSNR. Overall, Pawprint v2024b serves as a solid baseline for traditional blind image watermarks, and attains a performance

| Metric      | Pawprint v2024b | PixelSeal+BCH   | DWT-DCT-SVD   |
|-------------|-----------------|-----------------|---------------|
| Payload     | <b>100 bits</b> | <b>100 bits</b> | 64 bits       |
| PSNR        | <b>46.79 dB</b> | 46.12 dB        | 39.34 dB      |
| wPSNR       | <b>42.08 dB</b> | 41.42 dB        | 34.62 dB      |
| SSIM        | 0.9965          | 0.9967          | <b>0.9999</b> |
| Encode time | 1.15 s          | <b>0.02 s</b>   | 0.33 s        |
| Decode time | 0.086 s         | <b>0.017 s</b>  | 0.220 s       |
| Clean BER   | <b>0.0000</b>   | <b>0.0000</b>   | <b>0.0000</b> |
| Attack BER  | 0.0178          | <b>0.0102</b>   | 0.1967        |

Table 1: Aggregate statistics of the three watermarking methods.

| Attack         | Pawprint v2024b | PixelSeal+BCH | DWT-DCT-SVD   |
|----------------|-----------------|---------------|---------------|
| Gaussian noise | <b>0.0260</b>   | 0.0703        | 0.1485        |
| Speckle noise  | <b>0.0172</b>   | 0.0434        | 0.0477        |
| Blur           | <b>0.0000</b>   | <b>0.0000</b> | <b>0.0000</b> |
| Brightness     | 0.0100          | <b>0.0083</b> | 0.3292        |
| Sharpness      | <b>0.0000</b>   | <b>0.0000</b> | 0.0021        |
| Gamma          | 0.0029          | <b>0.0008</b> | 0.3361        |
| Histogram eq.  | 0.0878          | <b>0.0020</b> | 0.4554        |
| Median filter  | 0.0011          | <b>0.0000</b> | 0.0043        |
| Rotation       | 0.0356          | <b>0.0006</b> | 0.4905        |
| Crop 10%       | 0.0200          | <b>0.0010</b> | 0.4831        |
| Scale 0.5      | <b>0.0000</b>   | <b>0.0000</b> | <b>0.0000</b> |
| Flip H         | <b>0.0000</b>   | <b>0.0000</b> | 0.4774        |
| Resize 50%     | <b>0.0000</b>   | <b>0.0000</b> | <b>0.0000</b> |
| JPEG q=50      | 0.0004          | <b>0.0000</b> | 0.2311        |
| JPEG 2000      | <b>0.0037</b>   | 0.0045        | 0.0409        |
| JPEG XL        | 0.0031          | <b>0.0000</b> | 0.1237        |
| JPEG AI        | 0.0353          | <b>0.0094</b> | 0.3531        |
| AI remove      | 0.0780          | 0.0422        | <b>0.0161</b> |

Table 2: BER by attack for the benchmark settings used in Table 1.

that is good enough to use in commercial content provenance applications (i.e. imperceptible to the human eye while still durable against common image edits).

## 4 Conclusion

Pawprint v2024b pushes traditional methods to the limit, leapfrogging existing traditional image watermarks, and—after 8 years of innovation in deep learning methods coupled with recent investment in the technology by technology giants such as Meta, Google, and Adobe—still matches the performance of the SotA neural network watermarks of 2026. In the JPEG Trust Watermarking Benchmark ICIIP 2026 Grand Challenge, Pawprint v2024b achieves an average BER of under 2% at 100 bits and an average PSNR of 47.

The limit of neural network methods has not been reached yet; more improvements will come in the near future. We at Trufo Inc. are developing a suite of neural network watermarks, which we plan to publish in due course.

## References

- [1] I. J. Cox, J. Kilian, T. Leighton, and T. Shamon, “Secure spread spectrum watermarking for multimedia,” *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.
- [2] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, “HiDDen: Hiding data with deep networks,” in *Proc. European Conference on Computer Vision*, 2018.
- [3] M. Hamidi, M. El Haziti, H. Cherifi, and M. El Hassouni, “A hybrid robust image watermarking method based on DWT-DCT and SIFT for copyright protection,” *Journal of Imaging*, vol. 7, no. 10, 2021, doi: 10.3390/jimaging7100218.
- [4] T. Souček, P. Fernandez, H. El Sahar, S.-A. Rebuffi, V. Lacatusu, T. Tran, T. Sander, and A. Mourachko, “Pixel Seal: Adversarial-only training for invisible image and video watermarking,” 2025.
- [5] Google DeepMind, “SynthID: Identifying AI-generated images,” 2023. [Online]. Available: <https://deepmind.google/technologies/synthid/>
- [6] T. Bui, S. Agarwal, and J. Collomosse, “TrustMark: Robust watermarking and watermark removal for arbitrary resolution images,” in *Proc. IEEE International Conference on Computer Vision*, 2025.
- [7] Trufo Inc., “Solutions to the content authenticity problem,” 2024. [Online]. Available: <https://trufo.ai/resources/solutions-to-the-content-authenticity-problem.pdf>