NRGMark: Localized Watermarking for Energy Transparency in Images

Shruti Agarwal[1] Élie Michel[1] Vishal Asnani[1] Tania Mathern[1] John Collomosse[1,2]
[1]Adobe Research.
[2]DECaDE, University of Surrey.
{shragarw, emichel, vasnani, mathern, collomos}@adobe.com

Abstract

We present **NRGMark**, *a region-based image watermarking framework to embed provenance metadata into composite graphic designs such as posters. NRGMark enables imperceptible watermarking of distinct visual elements each carrying independent metadata on aspects like environmental impact, such as the energy consumption associated with generative AI (GenAI) use. NRGMark extends image watermark encoder-decoder models by incorporating an object localization network to detect and decode multiple watermarked regions within a document, even under image transformations and physical print–scan degradation. NRGMark interoperates with several watermarking techniques and the emerging C2PA open standard for media provenance to encode environmental impact metadata. We demonstrate NRGMark on both synthetic and real-world design layouts, illustrating its potential to support energy transparency in the age of GenAI.*

1. Introduction

As generative AI (GenAI) becomes increasingly integrated into creative workflows, so too does the environmental cost of large-scale model inference and training [27]. Energy transparency in digital content creation is an emerging requirement in responsible AI and sustainable media production (*e.g.* EU AI Act, Article 53 [21]). However, this energy footprint is typically opaque to end users. To support traceability goals and sustainability in GenAI, there is a growing need for methods that expose and persist metadata on environmental impact in a tamper-evident way [19].

Watermarking provides a promising mechanism for conveying such metadata, embedding information directly into the pixels of an image in a robust and imperceptible manner [18]. Yet, most watermarking systems encode only a single payload per image, either globally [15, 22, 46] or distributed across multiple regions [28]. This limits their applicability to composite designs (*e.g.* posters or advertisements) that comprise heterogeneous visual elements originating from different sources and incurring distinct energy costs.

In this paper, we propose NRGMark; a region-based watermarking method that enables *granular energy transparency* over images. NRGMark augments encoder–decoder watermarking models with an object localization network to detect and decode multiple watermarked regions in an image, such as a rendering, screenshot or scan of a printed design. Each region encodes a unique identifier linked to a provenance record describing the origin and environmental footprint of the corresponding visual element. These records are embedded using a custom extension to the emerging Coalition for Content Provenance and Authenticity (C2PA, ISO 22144) standard [17], ensuring interoperability and tamper-evident attribution. Our core technical contributions are:

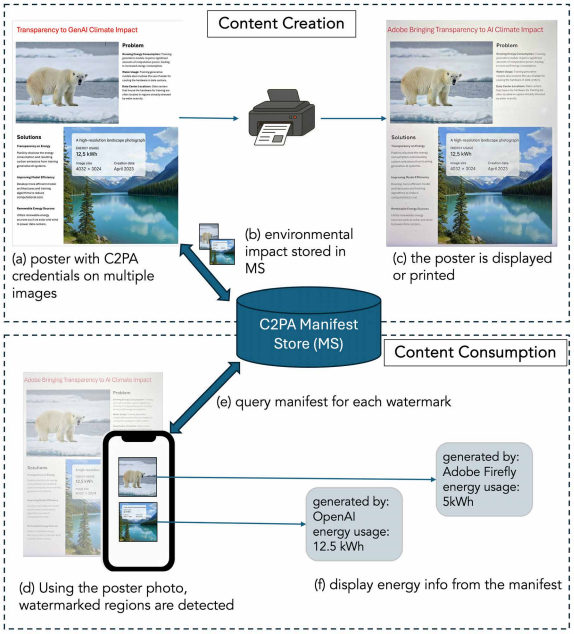

Figure 1. **NRGMark** watermarks visual elements of a graphic design, enabling lookup of environmental impact data, such as GenAI energy use. **Content Creation** (top): Region watermarking is used to embed unique ID in each element (a), linked to energy transparency data stored via the C2PA standard in a database (b). **Content Consumption** (bottom): Watermarks are detected in a printed version of the design (c,d), enabling recovery (e) and display (f) of image provenance and energy transparency data.

1. Region Proposal Network (RPN) to enable joint wa-

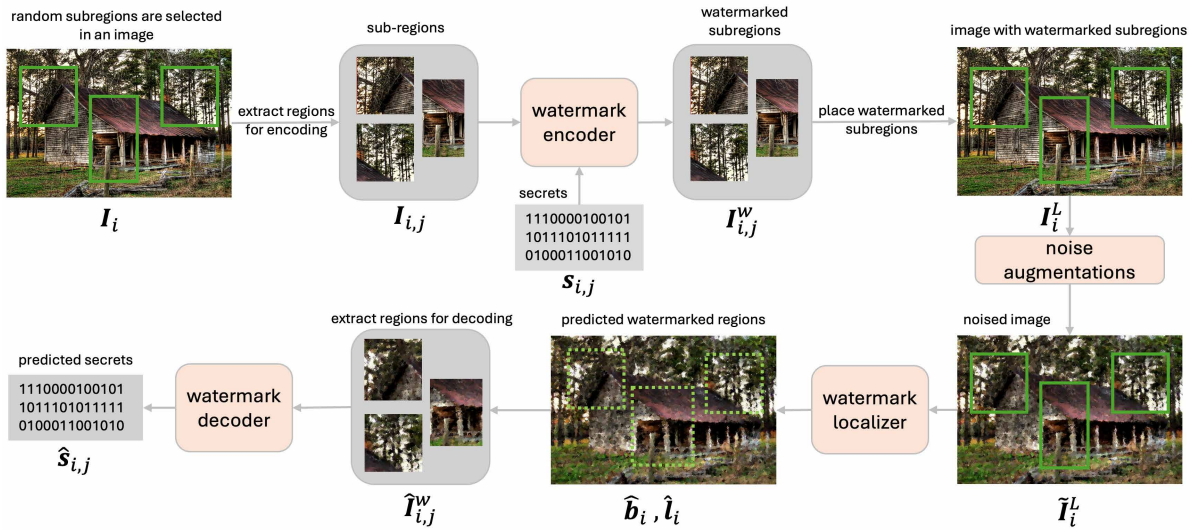

Figure 2. Overview of the NRGMark training pipeline. Given an input image I_i, a set of random subregions $I_{i,j}$ are selected and extracted. Watermarked regions are generated by embedding random binary secrets $s_{i,j}$ using a watermark encoder, producing $I^w_{i,j}$. These are composited back into the original image to form a region-level watermarked image I^L_i. The composite image is passed through a noise augmentation module to simulate real-world perturbations, resulting in a noised image $\tilde{I}^L_i$. The watermark localizer then predicts bounding boxes $\hat{b}_i$ and labels $\hat{l}_i$ for potential watermark regions, which are used to compute the localization loss. Subregions with positive predictions are extracted as $\hat{I}^w_{i,j}$ and decoded to retrieve secrets $\hat{s}_{i,j}$, which are compared to their ground-truth counterparts for computing the binary cross-entropy loss.

termark detection and localization via bounding box regression, supporting spatially precise recovery of multiple embedded payloads in both synthetic and real-world designs.

2. Training for watermark localization, demonstrating generality by extending several recent encoder–decoder watermarks [14, 15, 43, 46] to region-watermarking, using synthetic region generation and noise augmentation to ensure robustness under digital and physical transformations.

3. Extension of the C2PA provenance standard demonstrating its potential to link energy transparency data within images via 'soft binding' of region-level watermarks.

Even though we use energy transparency as an application of NRGMark, the ability to add localization to watermark detectors enables selective watermarking of image regions to embed copyright/ownership, GenAI labeling, license/consent, training/model provenance, safety notices or to avoid regions that exhibit visual artifacts enhancing the general quality of existing image watermarking methods via region-based watermarking.

2. Related Work

Media Provenance is the focus of the cross-industry C2PA standard [17], which encodes a metadata 'manifest' within assets (*e.g.* images) detailing their origin, edit history, and other contextual signals to support trust and creator attribution. Provenance has been explored for archival integrity [11, 12], AI training consent [6] and content licensing frameworks [5, 7, 8, 33]. Yet, energy transparency has not yet been addressed. We propose, for the first time, the use of provenance to carry this signal. Metadata is also easily stripped through printing, screenshots, or image reprocessing via social platforms media. Thus, content-aware technologies such as perceptual hashing [9, 32, 44] or watermarking methods [4, 20, 41] are used to identify assets and look up their C2PA metadata in a cloud [18, 34] or blockchain [10] based database. In NRGMark we go further to identify multiple distinct watermarks within an image, using these to lookup component-wise C2PA manifests which we extend to carry environmental impact data.

Image watermarking has been successfully applied to a range of use cases, including copyright protection [39], detection of manipulated or AI-generated media [1, 45], attribution [2, 3], and content provenance [15, 24]. A wide variety of watermark encoder–decoder architectures have been proposed to embed imperceptible signals into images, which can later be decoded into secret messages. Classical methods include least significant bit (LSB) substitution [42] and more robust spatial and frequency domain techniques such as DCT, DWT, and hybrid methods like DWT-DCT-SVD [25, 31, 37]. Several watermarking approaches have been developed specifically for flagging GenAI content, such as SynthID [26], Stable Signature [23], and TreeRing [40]. However, these are binary detectors rather than decoders, and do not recover message payloads linked to provenance data. Deep learning based watermarking, pioneered by HiDDeN [46], introduced encoder–decoder architectures capable of embedding identifiers into arbitrary image content. This direction has since evolved via StegaStamp [38], which improves geometric resilience; SSL [22] and RoSteALS [14], which operate in latent space; and TrustMark [13, 15], which introduced point convolutions

Figure 3. Training inputs to the watermark localizer module. A green box indicates a watermarked region; a red box indicates a non-watermarked region. Note in the bottom examples, watermarked regions may be composited with a distinct boundary or may be watermarked sub-regions within the background image.

and frequency-domain loss. InvisMark [43] adopts the losses and the architecture of TrustMark, but adds zero-padding to force the watermark to a single center squared region to increase PSNR. More recently, general region-based watermarking has been explored. WAM [36] predicts a segmentation mask for multiple watermarked regions in an image, but supports only 32-bit secrets and suffers reduced accuracy beyond three regions. Hu *et al*. [28] also targets arbitrary regions, but encodes only a single secret. Most existing watermarking methods assume a single secret, limiting their use in our scenario that requires multiple independent watermarks for distinct visual elements. Our proposed NRGMark method introduces a way to convert existing encoder–decoder watermarking architectures into region-based systems. This enables the embedding of multiple imperceptible watermarks, each carrying distinct provenance-linked identifiers, to support granular energy transparency in multi-element visual designs.

Object detection: Object detection has seen significant advances through deep learning, particularly with the adoption of convolutional neural networks (e.g., Faster R-CNN [35] and DETR [16]). These models enable reliable localization of semantic content within images, and in our context, form the foundation for detecting region-bound watermarks. Unlike global watermarking approaches, NRGMark leverages object detection models to learn the spatial distribution of watermark signatures. This enables retrieval of embedded environmental impact metadata with fine-grained localization, even in complex images or printed media using bounding box prediction.

3. Method

3.1. Background

Object Localization. Object detection methods such as Faster R-CNN use a Region Proposal Network (RPN) to identify and refine object locations. The RPN generates candidate bounding boxes by sliding a lightweight neural network over convolutional feature maps and regresses their coordinates to align with the ground-truth objects. This localization is trained using a multi-task loss that combines classification and regression:

$$L_{\text{rpn}} = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*), \tag{1}$$

where L_{cls} is a softmax log-loss for binary object classification with predicted scores p_i and labels $p_i^* \in \{0, 1\}$, and L_{reg} is a smooth L_1 loss between predicted box parameters t_i and ground-truth values t_i^*. The regression loss is only applied to positive anchors ($p_i^* = 1$). This ensures the network learns accurate localization and object confidence.

Watermark Encoder-Decoder. Modern image watermarking systems commonly include an encoder $\text{Enc}(\cdot)$, a decoder $\text{Dec}(\cdot)$, and a noise augmentation module. The encoder embeds a secret message $\boldsymbol{s}$ into an input image $\boldsymbol{I}$, producing a watermarked image $\boldsymbol{I}^w$:

$$\boldsymbol{I}^w = \text{Enc}(\boldsymbol{I}, \boldsymbol{s}). \tag{2}$$

To simulate real-world conditions, the watermarked image is passed through a noise module $\mathcal{N}(\cdot)$, yielding a perturbed version $\tilde{\boldsymbol{I}}^w$:

$$\tilde{\boldsymbol{I}}^w = \mathcal{N}(\boldsymbol{I}^w). \tag{3}$$

The decoder attempts to recover the secret from this noisy watermarked image:

$$\hat{\boldsymbol{s}} = \text{Dec}(\boldsymbol{I}^w). \tag{4}$$

To make the watermark robust to real-world transformations (e.g., compression, geometric distortions, or print–scan processes), a variety of noise augmentations are applied during training. These include photometric changes (e.g., brightness, contrast, JPEG artifacts), geometric changes (e.g., scaling, rotation), and domain shifts. The degree and diversity of these augmentations vary based on the robustness and imperceptibility goals of each watermarking system.

Energy Provenance. The Coalition for Content Provenance and Authenticity (C2PA) is a cross-industry standards group that has developed an open specification called *Content Credentials*, for encoding provenance metadata within images and other media [17]. The standard describes provenance within a metadata structure called a *manifest*. The manifest contains facts, called *assertions*, about the image such as who made it, when and how *i.e*. the *actions* were performed on the content. Assertions may also reference *ingredient* image(s) used to create the image, which may in turn also carry manifests and so recursively encode a provenance graph. The set of assertions for a given image is called a *claim* and is both cryptographically signed, and bound to the image pixels in a tamper-evident way using a cryptographic hash (*e.g*. SHA-256) called a *hard binding*.

Figure 4. Customizing C2PA to carry environmental impact data for NRGMark. Left: Graphic design comprising six commercially generated GenAI elements, each carrying environmental impact data. Content Credentials inspection tool visualizes NRGMark data, *e.g.* the energy use of a particular node in the provenance graph as a proportion of the total. The GenAI visual elements (ingredients) each carry impact data (bottom), recursively summed total over the graph (top); mock values are shown. Right: C2PA manifest for a visual element carries environmental impact data either by decorating each of the recorded actions (Option A, yellow) or by summarizing data for the entire manifest using an assertion (Option B, green). The examples shown (left) decorate the `c2pa.created` action (Option A).

Unfortunately, many content platforms strip C2PA metadata from images, as does screenshotting or printing. Thus, to enhance durability, imperceptible watermarks are often used in tandem with that metadata, to lookup a stripped manifest [18]. The watermark serves as an identifier or so-called *soft binding*, which acts as a key to perform search and recovery of the manifest. Due to its growing adoption we build on C2PA to carry environmental impact data.

3.2. NRGMark Pipeline

The NRGMark framework is designed to detect and localize multiple watermarked regions of arbitrary size within a single image. Each watermarked region carries its own secret, enabling fine-grained traceability. The pipeline consists of three core stages: (1) watermark region construction, (2) localization and detection, and (3) secret decoding. We describe each stage below.

Watermarked Region Construction At training time, we synthesize training data by assembling watermarked regions into mixed images. Given a batch of RGB images $\mathcal{I} = \{\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_n\}$, where each image $\boldsymbol{I}_i \in \mathbb{R}^{H\times W\times 3}$, we randomly select k subregions per image that may be watermarked. The construction process involves sampling the following for each image $\boldsymbol{I}_i$:

- Non-overlapping bounding boxes $\boldsymbol{b}_{i,j} \in \mathbb{R}^4$ for each region $j \in \{1, \ldots, k\}$, specifying the region corners as $(x_{\text{left}}, y_{\text{top}}, x_{\text{right}}, y_{\text{bottom}})$.
- Indices $c_{i,j} \in \{1, \ldots, n\}$ determining the source image from which content for subregion j is drawn. If $i == j$, the content is from the same image resulting in no-boundary watermark regions, shown in Fig. 3 bottom row.
- Labels $l_{i,j} \in \{0, 1\}$ that specify whether the subregion should be watermarked (1) or remain clean (0).

Using this information, each subregion is extracted from the appropriate image using a cropping function:

$$\boldsymbol{I}_{i,j} = \text{Extract}(\boldsymbol{I}_{c_{i,j}}, \boldsymbol{b}_{i,j}), \tag{5}$$

where $\text{Extract}(\cdot)$ returns the image patch corresponding to the bounding box $\boldsymbol{b}_{i,j}$ from source image $\boldsymbol{I}_{c_{i,j}}$.

For each subregion labeled to be watermarked ($l_{i,j} = 1$), a binary secret $\boldsymbol{s}_{i,j} \in \{0, 1\}^l$ is randomly generated. This secret is embedded into the subregion using the encoder:

$$\boldsymbol{I}^w_{i,j} = \text{Enc}(\boldsymbol{I}_{i,j}, \boldsymbol{s}_{i,j}), \tag{6}$$

producing the watermarked patch $\boldsymbol{I}^w_{i,j}$. Subregions not marked for watermarking are kept as-is. These watermarked and non-watermarked patches are then reassembled into the original image to construct a composite image $\boldsymbol{I}^L_i$ with mixed watermark presence:

$$\boldsymbol{I}^L_i = \text{Assemble}(\boldsymbol{I}_i, \{(\boldsymbol{b}_{i,j}, l_{i,j}, \boldsymbol{I}_{i,j} \text{ or } \boldsymbol{I}^w_{i,j})\}_{j=1}^k). \tag{7}$$

This composition simulates a real-world setting in which only certain regions of an image carry embedded provenance data (Fig. 3).

Watermark Region Localization To improve the robustness to real-world distortions, the composite image $\boldsymbol{I}^L_i$ is first passed through a noise augmentation module:

$$\tilde{\boldsymbol{I}}^L_i = \mathcal{N}(\boldsymbol{I}^L_i), \tag{8}$$

MIRFlickR [29]								
	No Noise				With Noise			
NRGMark (methods)	Precision	Recall	Detection Accuracy	IoU	Precision	Recall	Detection Accuracy	IoU
Trustmark-P [15]	0.96	0.97	1.00	0.95	0.90	0.91	1.00	0.94
Trustmark-Q [15]	0.97	0.97	1.00	0.93	0.92	0.93	1.00	0.92
RoSteALS [14]	1.00	1.00	0.99	0.99	0.90	0.89	0.97	0.96
InvisMark [43]	0.99	0.99	1.00	0.95	0.91	0.90	1.00	0.92
HiDDeN [46]	1.00	1.00	1.00	0.99	0.98	0.98	1.00	0.98
COCO [30]								
	No Noise				With Noise			
NRGMark (methods)	Precision	Recall	Detection Accuracy	IoU	Precision	Recall	Detection Accuracy	IoU
Trustmark-P [15]	0.98	0.98	1.00	0.87	0.93	0.94	1.00	0.87
Trustmark-Q [15]	0.97	0.98	1.00	0.84	0.94	0.96	1.00	0.81
RoSteALS [14]	0.94	0.97	0.99	0.96	0.82	0.85	0.95	0.91
InvisMark [43]	0.98	0.98	1.00	0.93	0.96	0.96	1.00	0.90
HiDDeN [46]	0.94	0.95	1.00	0.96	0.92	0.93	1.00	0.95

Table 1. Detection and localization accuracy of NRGMark when used to add region watermarking capabilities to five watermarking encoder-decoder models. Evaluation performed over MIRFlickR (top) and COCO (bottom) datasets.

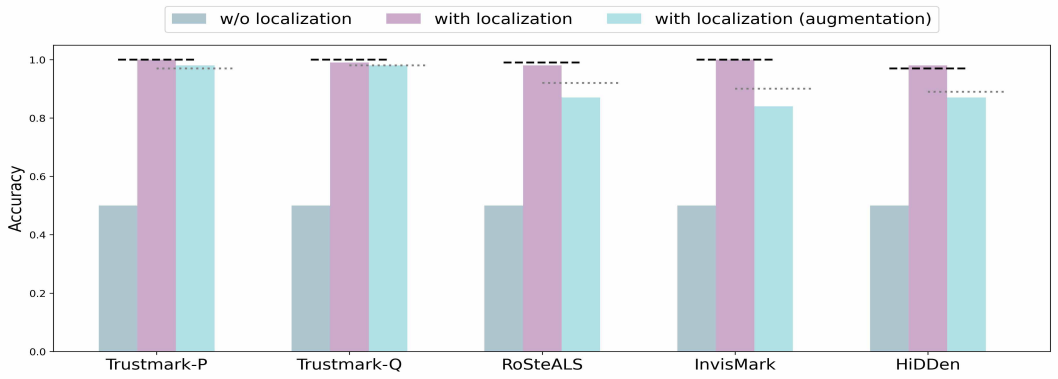

Figure 5. Comparison of bit accuracy before and after localization for the clean and noisy images (MIRFlickR dataset).

where $\mathcal{N}(\cdot)$ applies transformations such as JPEG compression, brightness shifts, scaling, rotation, or print–scan simulation. If geometric transformations are included, the original bounding boxes $\boldsymbol{b}_{i,j}$ are also transformed accordingly to maintain alignment with the visual content.

The resulting noisy image $\tilde{\boldsymbol{I}}_i^L$ is then inputted into the localizer module, which predicts a set of bounding boxes $\hat{\boldsymbol{b}}_i = \{\hat{\boldsymbol{b}}_{i,1}, \ldots, \hat{\boldsymbol{b}}_{i,\hat{k}}\}$ and their associated labels $\hat{\mathbf{l}}_i = \{\hat{l}_{i,1}, \ldots, \hat{l}_{i,\hat{k}}\}$. The localizer is trained using a region proposal network (RPN) loss, as follows:

$$\hat{\boldsymbol{b}}_i, \hat{\mathbf{l}}_i, L_{\text{rpn}} = \text{Localizer}(\tilde{\boldsymbol{I}}_i^L, \boldsymbol{b}_i, \mathbf{l}_i), \tag{9}$$

where L_{rpn} is defined in Eq. 1. During inference, the localizer operates without access to $\boldsymbol{b}_i$ and $\mathbf{l}_i$, relying solely on $\tilde{\boldsymbol{I}}_i^L$ to make predictions.

Secret Decoding and Loss Computation To recover the embedded secrets, we first extract the predicted regions marked as watermarked:

$$\hat{\boldsymbol{I}}_{i,m}^w = \text{Extract}(\tilde{\boldsymbol{I}}_i^L, \hat{\boldsymbol{b}}_{i,m}) \quad \text{for all } \hat{l}_{i,m} = 1. \tag{10}$$

Each predicted region $\hat{\boldsymbol{I}}_{i,m}^w$ is passed through the decoder to obtain a recovered secret:

$$\hat{\boldsymbol{s}}_{i,m} = \text{Dec}(\hat{\boldsymbol{I}}_{i,m}^w). \tag{11}$$

To compute the secret loss, we first identify the ground-truth secret $\boldsymbol{s}_{i,j}$ corresponding to the region with highest Intersection-over-Union (IoU) overlap with $\hat{\boldsymbol{b}}_{i,m}$. The binary cross-entropy loss is then computed as:

$$L_{\text{BCE}}(\boldsymbol{s}_{i,m}, \hat{\boldsymbol{s}}_{i,m}) = -\frac{1}{l}\sum_{j=1}^{l}\left[s_j \log(\hat{s}_j) + (1 - s_j)\log(1 - \hat{s}_j)\right], \tag{12}$$

where l is the length of the secret bit string. This loss penalizes incorrect bit predictions and ensures accurate recovery of embedded messages.

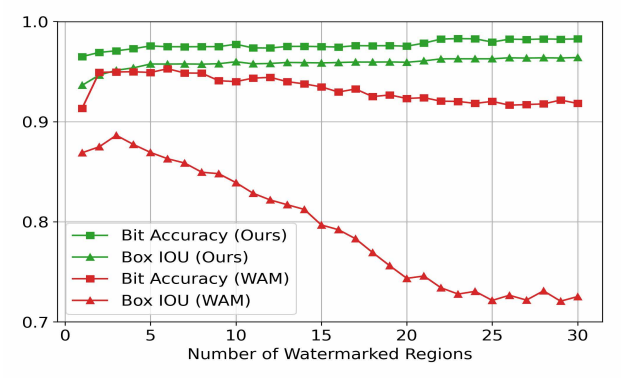

Figure 6. Comparison of NRGMark and WAM [36] with 1 to 30 watermarked regions in an image.

Finally, the overall training loss is a weighted combination of the localization and secret decoding losses:

$$L = L_{\text{rpn}} + L_{\text{BCE}}. \tag{13}$$

This end-to-end objective enables the system to jointly learn where watermarks exist in an image and how to decode their payloads, even under distortion.

Watermark Lookup Each watermark embeds random unique identifier (UUID) which serves as a key to lookup C2PA metadata (manifests) in a cloud-based C2PA Manifest Repository (MR). The MR is a key-value store, implemented using DynamoDB. It is accessed via the standardized C2PA Soft Binding Resolution (REST) API enabling federated search for manifests using watermark IDs [17], following the system architecture described in [18].

Energy Provenance NRGMark proposes to decorate the C2PA provenance graph with environmental impact data, by introducing a custom assertion and modifiers to actions encoded in the C2PA manifest. Each visual element (ingredient) may be tagged directly with an *assertion* describing its energy footprint (Fig. 4 right) using a data schema that currently supports: 1) greenhouse gas emission (kilograms of CO_2 equivalent); 2) water use (liters); and 3) energy use data (kilowatt-hours). Alternatively, *actions* performed on ingredients may each be decorated via the same data schema. Fig. 4 (right) shows examples of both options. The methods by which data may be collected per visual element vary across AI implementors. In our experiments, impact metrics are treated as externally supplied annotatoins, obtained via either (i) direct metering of training/inference (*e.g.* PowerAPI) or (ii) estimation from hardware specs, runtime, and cloud-provider model-card disclosures. Environmental impact data for an entire asset may be evaluated by recursively summing the relevant impact data fields across the full provenance graph, or for sub-graphs or even individual visual elements. Fig.4 shows this functionality embedded in a commercial C2PA inspection app.

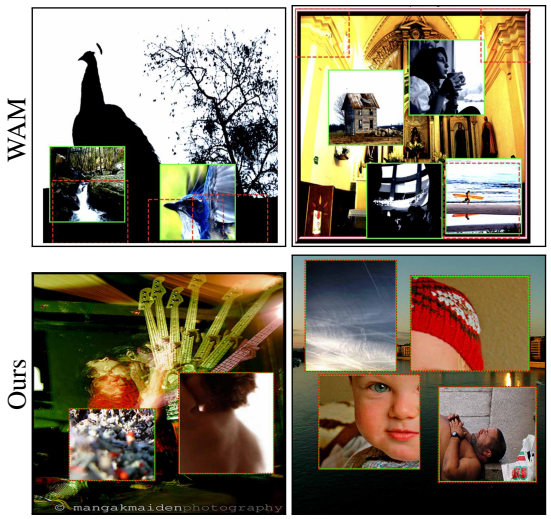

Figure 7. Posters created using WAM [36] (top) and NRGMark (bottom) with 1-4 watermarked subregions. The green box are the ground-truth watermark regions and red dotted boxes are the predictions by WAM or NRGMark localizer.

4. Experiments

Implementation Details As we train a separate watermark localization module for each of the watermarking techniques, here we describe the training parameters that are common and the ones that are specific to the watermarking techniques. For training the Localizer(.) for any watermarking technique, we adopt the same training parameters, like secret length, learning rate, noise augmentations, as were used for that method. For each training, the watermark encoder and decoder are kept frozen while only training the localizer module. Unless specified otherwise, the localizer is trained with 1 to 4 watermarking regions, i.e. $k \in [1, 4]$. The background image is of size 800 to 1500 and each watermark subregion is a non-square image that can cover 10% to 90% of the background image on each side which corresponds to subregions of 150px to 1350px in the background image. We train each version of our NRGMark pipeline on 100K images from the MIRFlickR 1M dataset [29], and then evaluate it using 10K other images from MIRFlickR. To assess generalization to natural scene objects, we also evaluate NRGMark on the first 10K validation images of COCO [30]. For COCO, we select images that contains 1-4 bounding boxes of size at least 10% of the image.

Evaluation Metrics We evaluate the localizer using five key metrics: *precision, recall, detection accuracy, IoU, and bit accuracy*. All metrics are computed on predictions with a confidence score greater than 0.9, ensuring only high-confidence detections are considered.

IoU and *bit accuracy* are calculated only for predictions labeled as watermarked (i.e., predicted label = 1); non-watermarked predictions are excluded from these two metrics. *Precision* and *recall* are computed based on the number of true positives, false positives, and false negatives,

Figure 8. Printed and scanned example pages with four watermarked images encoded using Trustmark-Q model. Marked with green are the bounding box detected by NRGMark.

where a true positive is defined as a predicted watermarked region that overlaps a ground-truth watermark region with an IoU of at least 0.9.

Detection accuracy measures the proportion of predicted labels that correctly match the label of the ground-truth region with the highest IoU. *Bit accuracy* is computed only for predicted watermarked regions, by comparing the decoded secret to the ground-truth secret associated with the most overlapping ground-truth box.

4.1. Results

Localization Performance Across Watermarking Techniques We evaluate our watermark localizer across five different watermarking techniques and two datasets—MIRFlickr (top) and COCO (bottom)—under both clean and noisy conditions, as summarized in Tab. 1. Among these methods, Trustmark-P and Trustmark-Q refer to two variants of the Trustmark framework [15], optimized respectively for visual quality and robustness. Across all methods, our localizer achieves near-perfect detection accuracy, even when applied to perturbed images. Details of the noise settings used for each technique are provided in the Sup. Mat..

Using the predicted bounding boxes, we extract the watermarked regions and decode them using the original decoder associated with each watermarking technique. Fig. 5 presents bit accuracy in three settings for each technique: (i) decoder-only (no localization), (ii) with localization on clean images, and (iii) with localization on noisy images. The horizontal dashed lines represent the baseline bit accuracy achieved by each decoder when provided with ground-truth watermarked regions. Our localizer recovers bit accuracy comparable to the baseline in the clean setting. Under noise, bit accuracy remains close to the baseline for Trustmark-P, Trustmark-Q, and HiDDeN. However, we observe a noticeable drop for RoSteALS and InvisMark, which we attribute to their sensitivity to bounding box padding and misalignment during region extraction.

Comparison with Region-Based Watermarking Method We compare NRGMark against WAM [36], a recent region-based watermarking technique that can embeds multiple localized watermarks into an image. We re-train NRGMark with Trustmark-Q model of 32-bit payload to match with WAM and compare both methods on bit accuracy and box IoU. For WAM, we used the minimum enclosing bounding box of predicted watermarked segment. As the WAM and Trustmark-Q decoders accept different input sizes (512×512 and 256×256, respectively), we choose patch sizes that keep the same side-length ratio 12.5% relative to the background in both cases. Concretely, for WAM we use 64×64 patches on 512×512 images, and for NRGMark we use 256×256 sub-regions on 2048×2048 backgrounds. As shown in Fig. 6, our method consistently outperforms WAM in both box IoU and bit accuracy across all region counts. Additionally, NRGMark remains stable beyond three regions, while WAM's performance drops.

Figure 9. Representative watermark detection on printed/scanned poster, with per component environmental impact extracted from the retrieved provenance records (C2PA manifests).

Additionally, we evaluated WAM similar to Table 1. To create a poster like Fig. 3, 1-4 images of size 512×512 were watermarked using WAM and placed randomly on a background image of size 1600×1600, see Fig. 7 for examples. This composite image was then resized back to 512×512 and evaluated using WAM decoder. The precision, recall, bit accuracy, and box IoU of WAM are 0.20, 0.10, 0.31, 0.86. The low precision, recall, and box IoU indicates that WAM decoder fails to localize the correct location and count of watermark subregions (Fig. 7).

Application to Energy Provenance We demonstrate how NRGMark can be used to track energy information across the image asset's supply chain in a document composed of multiple watermarked images. First, we quantify the performance of NRGMark in detecting multiple watermarked regions from a printed poster. We print 40 images simulating a poster-like setting, where a US Letter-sized paper displays 1-4 images watermarked using Trustmark-Q. Shown in Fig. 8 are some example print/scan images with bounding box detections shown in green. These images are printed and scanned using a Canon Image Runner Advance 3530i printer and scanner. For each image, we detect and decode the watermarks, achieving an average bit accuracy of 0.91

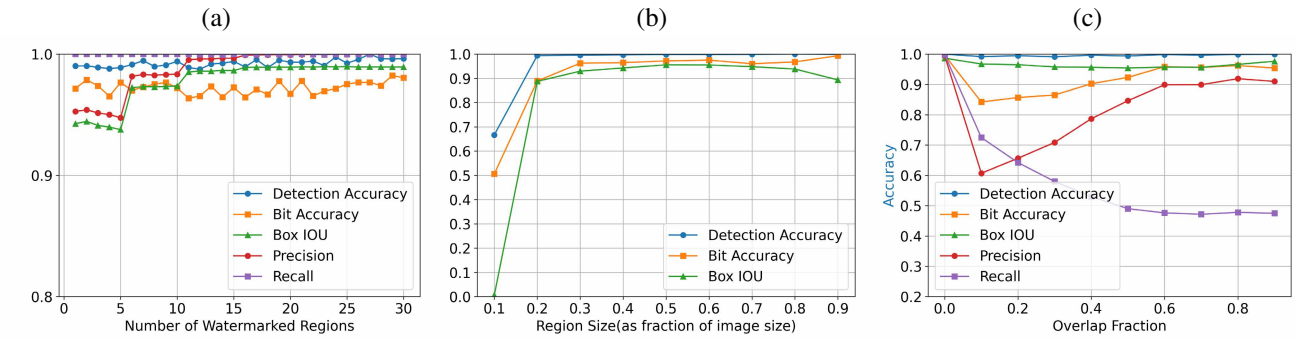

Figure 10. Ablations on number of regions, size of the regions, and overlapping regions.

with a standard deviation of 0.08, correctly identifying 80% of the watermarks using BCH error correction with 40 bits. Fig. 9 shows an example of multiple watermarks being used to retrieve provenance, providing transparency about the environmental impact of the images. Additional print robustness against text distraction, size variations, and clean images are provided in the Sup. Mat..

4.2. Ablations

Number of Regions Fig. 10a shows detection accuracy, IoU, and bit accuracy as the number of regions in the image varies from 1 to 30. We train our model using Trustmark-Q watermarking on background images of size 2048×2048. During evaluation, the watermark sub-region size is fixed to 256×256, matching the training resolution of Trustmark. Localization accuracy remains largely unaffected by the number of watermarked regions.

Size of Region Fig. 10b shows performance for different watermark region sizes, ranging from 10% to 90% of a background image sized 1500×1500. For this experiment, we fix the number of watermark sub-regions to one. Detection accuracy and IoU are lowest for 10% region size but exceed 90% once the watermark region size increases to 20% or more. Failure cases are shown in the Sup. Mat. We hypothesize that the Trustmark watermark becomes weak at 150px resolution, as it is trained for 256×256, making it difficult for the localizer to detect.

Region Overlap Fig. 10c shows the localizer and decoder performance when two watermark regions overlap from 10% to 50%. For this experiment, we fix the number of watermark regions to two, each sized 256×256. Although detection accuracy and IoU are unaffected by the overlap, bit accuracy and precision initially drop as overlap increases, then recover. This is explained by the recall, which drops to around 0.5 when overlap exceeds 40%, indicating that only one of the two regions is detected. This is expected, as Trustmark-Q is trained to handle up to 20% occlusion.

Training without non-watermark regions We retrained NRGMark without any non-watermarked images, see the red boxes in Fig. 3. The evaluation is done on MIRFlickR dataset for Trustmark-Q model with noise. As shown in Tab.2, the precision and detection accuracy collapse to near chance, because the model w/o non-watermark training frequently misclassifies clean regions as watermarked.

NRGMark (methods)	Precision	Recall	Detection Accuracy	IoU
with	0.92	0.93	1.00	0.92
w/o	0.66	0.96	0.52	0.93

Table 2. Performance of NRGMark with Trustmark-Q when trained with (top) or without (bottom) the non-watermark images, see the red boxes in Fig. 3.

5. Conclusion

We introduced NRGMark, a region-based watermarking framework that supports fine-grained provenance and energy transparency within composite visual designs. By extending contemporary encoder–decoder watermarking models with localization, NRGMark enables the robust embedding and recovery of multiple independent watermarks under digital noise and print degradation. These watermarks act as soft bindings to C2PA manifests; standardized provenance records that we extend to express environmental impact data. We demonstrate that NRGMark generalizes across several watermarking techniques, maintains high detection and decoding accuracy under distortion, and scales to realistic poster-style layouts. To the best of our knowledge, we are the first to apply watermarking to energy transparency in digital media production, or to consider the environmental impact of digital assets as part of their media provenance. As the environmental implications of GenAI become more prominent [21], we see NRGMark as a step toward embedding sustainability into the visual media supply chain, ensuring impact data is declared and durably bound to creative outputs. Future work could include advocacy for adoption of environmental impact data assertions within the C2PA standards ecosystem (*e.g.* as a community extension) and exploration of other modalities.

Source Code: The bounding box detectoin models for Trustmark-P and Trustmark-Q from NRGMark are released within the MIT licensed Trustmark codebase at `https://github.com/adobe/trustmark`

References

[1] Vishal Asnani, Xi Yin, Tal Hassner, Sijia Liu, and Xiaoming Liu. Proactive image manipulation detection. In *CVPR*, 2022. 2

[2] Vishal Asnani, John Collomosse, Tu Bui, Xiaoming Liu, and Shruti Agarwal. Promark: Proactive diffusion watermarking for causal attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10802–10811, 2024. 2

[3] V. Asnani, J. Collomosse, X. Liu, and S. Agarwal. Custommark: Customization of diffusion models for proactive attribution. In *Proc. ICCV Workshop on Authenticity & Provenance in the age of Generative AI (APAI)*, 2025. 2

[4] S. Baba, L. Krekor, T. Arif, and Z. Shaaban. Watermarking scheme for copyright protection of digital images. *IJCSNS*, 9(4), 2019. 2

[5] Kar Balan, Shruti Agarwal, Simon Jenni, Andy Parsons, Andrew Gilbert, and John Collomosse. EKILA: Synthetic media provenance and attribution for generative art. In *CVPR*, 2023. 2

[6] K. Balan, A. Black, S. Jenni, A. Gilbert, A. Parsons, and J. Collomosse. DECORAIT - DECentralized Opt-in/out Registry for AI Training. In *Conference on Visual Media Production (CVMP), 2023*, 2023. 2

[7] K. Balan, A. Gilbert, and J. Collomosse. Content ARCs: Decentralized Content Rights in the Age of Generative AI. In *International Conference on AI and the Digital Economy (CADE), 2025*, 2025. 2

[8] J. Bennett, J. Collomosse, R. Gregory-Clarke, J. Jones, L. Love, M. Lycett, and W. Saunders. Time to ACCCT: Providing Creative Industries and AI Developers with a Copyright Framework of Access, Control, Consent, Compensation and Transparency. CoSTAR/DECaDE/Sheridans Technical Report, 2025. 2

[9] A. Bharati, D. Moreira, P. Flynn, A. de Rezende Rocha, K. Bowyer, and W. Scheirer. Transformation-aware embeddings for image provenance. *IEEE Trans. Info. Forensics and Sec.*, 16:2493–2507, 2021. 2

[10] T. Bui, D. Cooper, J. Collomosse, M. Bell, A. Green, J. Sheridan, J. Higgins, A. Das, J. Keller, O. Thereaux, and A. Brown. Archangel: Tamper-proofing video archives using temporal content hashes on the blockchain. In *Proc. CVPR Workshop CV, AI and Blockchain*, 2019. 2

[11] T. Bui, D. Cooper, J. Collomosse, M. Bell, A. Green, J. Sheridan, J. Higgins, A. Das, J. Keller, O. Thereaux, and A. Brown. ARCHANGEL: Tamper-proofing Video Archives using Temporal Content Hashes on the Blockchain. In *CVPR Workshops (Computer Vision, AI and Blockchain), 2019*, 2019. 2

[12] T. Bui, D. Cooper, J. Collomosse, M. Bell, A. Green, J. Sheridan, J. Higgins, A. Das, J. Keller, and O. Thereaux. Tamper-proofing Video with Hierarchical Attention Autoencoder Hashing on Blockchain. *IEEE Transactions on Multimedia (TMM), 2020*, 2020. 2

[13] Tu Bui, Shruti Agarwal, and John Collomosse. Trustmark: Universal watermarking for arbitrary resolution images. *ArXiv e-prints*, 2023. 2

[14] Tu Bui, Shruti Agarwal, Ning Yu, and John Collomosse. RoSteALS: Robust steganography using autoencoder latent space. In *CVPR*, 2023. 2, 5

[15] Tu Bui, Shruti Agarwal, and John Collomosse. Trustmark: Robust watermarking and watermark removal for arbitrary resolution image. In *Proc. ICCV*, 2025. 1, 2, 5, 7

[16] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, pages 213–229, 2020. 3

[17] Coalition for Content Provenance and Authenticity. Technical specification 2.2. Technical report, C2PA, 2025. 1, 2, 3, 6

[18] J. Collomosse and A. Parsons. To Authenticity, and Beyond! Building Safe and Fair Generative AI upon the Three Pillars of Provenance. *IEEE Computer Graphics and Applications (IEEE CG&A)*, 2024. 1, 2, 4, 6

[19] Maximilian Dauner and Gudrun Socher. Energy costs of communicating with ai. *Frontiers in Communication*, 10: 1572947, 2025. 1

[20] P. Devi, M. Venkatesan, and K. Duraiswamy. A fragile watermarking scheme for image authentication with tamper localization using integer wavelet transform. *J. Computer Science*, 5(11):831–837, 2019. 2

[21] European Parliament and Council. Regulation (EU) 2024/1689 of the european parliament and of the council — artificial intelligence act. Official Journal of the European Union, 2024. Annex XI (energy consumption disclosure) to Article 53; available online. 1, 8

[22] Pierre Fernandez, Alexandre Sablayrolles, Teddy Furon, Hervé Jégou, and Matthijs Douze. Watermarking images in self-supervised latent spaces. In *Proc. ICASSP*, pages 3054–3058. IEEE, 2022. 1, 2

[23] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *ICCV*, 2023. 2

[24] Pierre Fernandez, Hady Elsahar, I. Zeki Yalniz, and Alexandre Mourachko. VideoSeal: Open and efficient video watermarking. *ArXiv e-prints*, 2024. 2

[25] Kazem Ghazanfari, Shahrokh Ghaemmaghami, and Saeed R Khosravi. Lsb++: An improvement to lsb+ steganography. In *TENCON 2011-2011 IEEE Region 10 Conference*, pages 364–368. IEEE, 2011. 2

[26] Google DeepMind. Identifying AI-generated images with SynthID. https://deepmind.google/discover/blog/identifying-ai-generated-images-with-synthid/, 2023. 2

[27] Jens Gröger, Felix Behrens, Peter Gailhofer, and Inga Hilbert. Environmental impacts of artificial intelligence. Technical report, Öko-Institut on behalf of Greenpeace Germany, Hamburg, Germany, 2025. 55 pages. 1

[28] Runyi Hu, Jie Zhang, Shiqian Zhao, Nils Lukas, Jiwei Li, Qing Guo, Han Qiu, and Tianwei Zhang. Mask image watermarking. *arXiv preprint arXiv:2405.11135*, 2025. 1, 3

[29] Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proc. ICMIR*, pages 39–43, 2008. 5, 6

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755, 2014. 5, 6

[31] KA Navas, Mathews Cheriyan Ajay, M Lekshmi, Tampy S Archana, and M Sasikumar. DWT-DCT-SVD based watermarking. In *COMSWARE'08*, pages 271–274. IEEE, 2008. 2

[32] Eric Nguyen, Tu Bui, Vishy Swaminathan, and John Collomosse. OSCAR-Net: Object-centric scene graph attention for image attribution. In *ICCV*, 2021. 2

[33] P. Rixhon. An update on JPEG trust. https://cawg.io/meeting-notes/_attachments/2025-01-21/jpeg-trust-presentation.pdf, 2025. 2

[34] A. Petrov, S. Agarwal, P. Torr, A. Bibi, and J. Collomosse. On the Coexistence and Ensembling of Watermarks. In *Intl. Conf. Neural Information Processing Systems (NeurIPS), 2025*, 2025. 2

[35] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 3

[36] Tom Sander, Pierre Fernandez, Alain Oliviero Durmus, Teddy Furon, and Matthijs Douze. Watermark anything with localized messages. In *Proc. ICLR*, 2025. 3, 6, 7

[37] Mustafa Sabah Taha, Mohd Shafry Mohd Rahem, Mohammed Mahdi Hashim, and Hiyam N Khalid. High payload image steganography scheme with minimum distortion based on distinction grade value method. *Multimedia Tools and Applications*, 81(18):25913–25946, 2022. 2

[38] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Proc. CVPR*, pages 2117–2126, 2020. 2

[39] Wenbo Wan, Jun Wang, Yunming Zhang, Jing Li, Hui Yu, and Jiande Sun. A comprehensive survey on robust image watermarking. *Neurocomputing*, 2022. 2

[40] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[41] Xinyu Weng, Yongzhi Li, Lu Chi, and Yadong Mu. High-capacity convolutional video steganography with temporal residual modeling. In *Proc. ICMR*, pages 87–95, 2019. 2

[42] Raymond B Wolfgang and Edward J Delp. A watermark for digital images. In *Proc. ICIP*, pages 219–222. IEEE, 1996. 2

[43] Rui Xu, Mengya Hu, Deren Lei, Yaxi Li, David Lowe, Alex Gorevski, Mingyu Wang, Emily Ching, Alex Deng, et al. Invismark: Invisible and robust watermarking for ai-generated image provenance. In *Proc. Winter Conf. on Appl. of Computer Visoin (WACV)*, 2025. 2, 3, 5

[44] X. Zhang, Z. H. Sun, S. Karaman, and S.F. Chang. Discovering image manipulation history by pairwise relation and forensics tools. *IEEE J. Selected Topics in Signal Processing.*, 14(5):1012–1023, 2020. 2

[45] Xuanyu Zhang, Zecheng Tang, Zhipei Xu, Runyi Li, Youmin Xu, Bin Chen, Feng Gao, and Jian Zhang. Omniguard: Hybrid manipulation localization via augmented versatile deep image watermarking. In *Proc. CVPR*, 2025. 2

[46] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In *Proc. ECCV*, pages 657–672, 2018. 1, 2, 5