THE JPEG BLIND SPOT: EXPOSING A CRITICAL VULNERABILITY IN DOCUMENT TAMPERING DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Current state-of-the-art document tampering detection models predominantly derive their success from a reliance on low-level JPEG compression artifacts, particularly Block Artifact Grids (BAG), to localize forged regions. In this paper, we expose a critical vulnerability inherent in this approach. We introduce novel BAG-aware adversarial attacks for document forgery that are designed to exploit the local statistical properties of these artifacts. When evaluated on the largest available document tampering benchmark, DocTamper, this attack catastrophically fools existing methods, reducing their detection rate to no better than random chance. This catastrophic failure reveals that these models fail to learn genuine semantic representations of tampering and instead rely on highly superficial and easily bypassed compression artifacts. Our work demonstrates a fundamental fragility in current document forensic systems and underscores the urgent need for robustness against such adversarial failures in security-critical applications.

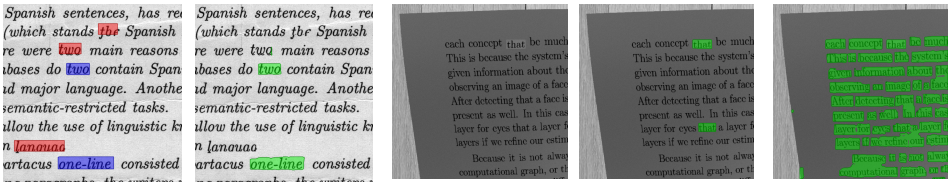
1 INTRODUCTION

Document integrity is critical in high-stakes domains such as finance, government administration, and academia, where even minor data manipulations by malicious actors can lead to serious information security risks (Verdoliva, 2020). At the same time, the rapid progress and widespread availability of modern image editing technologies have made it increasingly convenient to create such forgeries, necessitating efficient and robust methods for forgery detection (Nandanwar et al., 2021; Pun et al., 2023; Wu et al., 2019).

Deep learning (DL) has recently emerged as the de facto standard for document forgery detection (Qu et al., 2023; Wang et al., 2022b; Riaz et al., 2025; Chen et al., 2025), achieving state-of-the-art (SotA) performance on standard benchmarks (Qu et al., 2023; Wang et al., 2022b). However, despite significant architectural improvements over their predecessors, most existing SotA DL-based methods still largely rely on exploiting the frequency-domain artifact traces introduced by JPEG compression, particularly discontinuities in the block artifact grids (BAGs) (Li et al., 2009), as discriminative cues for detecting manipulated regions in the image.

While the above strategy is well-motivated, given that JPEG is a widely adopted compression algorithm for storing images, we hypothesize that an over-reliance on frequency-domain traces introduces a critical vulnerability in existing DL-based methods. JPEG compression operates on non-overlapping 8×8 blocks, with discrete cosine transform (DCT) coefficients computed independently for each block. These block-level features are then fed as input to the model in modern DL-based document forgery detection methods as an additional modality apart from the image features in the RGB-space (Qu et al., 2023; Riaz et al., 2025; Chen et al., 2025). We hypothesize that the Block-wise DCT features induces biases in detectors towards JPEG-grid distortions. for detecting forged regions, such as by simply validating the local block-level statistics (similar to the previous works Li et al. (2009); Nikoukhah et al. (2020) which detect forgeries by detecting block-level grid validity) rather than learning semantically meaningful, global representations of tampering.

To validate this hypothesis, we design two complementary adversarial attacks. First, a *BAG-aware copy–move forgery* preserves local JPEG block statistics by extracting and replacing text patches aligned to the 8 x 8 grid, thereby maintaining block artifact consistency across forged regions (see Fig. 1a). Second, a *Pad-Recompress-Crop (PRC)* attack deliberately shifts the JPEG grid via pad–recompress–crop, misaligning block boundaries with text glyphs and inducing atypical ringing ar-



(a) Attack 1: GACM (b) Attack 2: PRC

Figure 1: Examples show the effectiveness of our proposed forgery attacks. (a) The Grid-Aligned Copy–Move (GCAM)-based forgeries (left, red) preserves local JPEG block statistics compared to the misaligned case (blue), ensuring BAG consistency and evading frequency-centric detectors as can be seen from the predicted masks green (right). (b) The Pad–Recompress–Crop (PRC) attack distorts the grid-alignment without making any visible changes to the input (left) but causes model failures as shown by the predictions on the image before (middle) and after (right) the attack is applied.

tifacts (see Fig. 1b). Together, these attacks expose the shortcut bias of existing models, which fail to generalize beyond block-level cues and suffer widespread false positives under grid shifts. Intuitively, if the models had learned robust global feature representations beyond the block-level statistics, they should generalize over such manipulations; however, we show that this does not hold true in existing SotA methods.

We evaluate our proposed forgery method on one of the largest available document tampering benchmarks, DocTamper Qu et al. (2023), against several state-of-the-art deep learning-based document tampering detection methods. Our results show that, with only a simple exploit of grid alignment, our method can catastrophically fool existing SotA models, reducing their detection rates to as low as 1% in some cases. Moreover, we demonstrate that this vulnerability can also be leveraged to deliberately trigger false positives in existing methods, rendering them unreliable for deployment in high-stakes domains.

The main contributions of this work are following:

- We introduce two complementary, adversarial tampering procedures targeting JPEG block-grid shortcuts: (i) a BAG-aware, grid-aligned copy–move that preserves local JPEG statistics, and (ii) a pad–recompress–crop (PRC) attack that deliberately shifts the block-artifact grid without making any visible forgeries in the image.
- We evaluate our attacks on standard benchmarks and demonstrate that existing state-of-the-art DL-based document forgery detectors can fail catastrophically under these attacks (down to 1% detection) and that grid shifts can be used to trigger systematic false positives.

2 Related Work

JPEG Forensics JPEG is the most widely adopted format for compressed images, and forensic analysis based on its artifacts has a long history. Early works focused on detecting double JPEG compression (Wang & Zhang, 2016; Fan & de Queiroz, 2003) and were later extended to tampering localization (Barni et al., 2010; Chen & Hsu, 2008; Li et al., 2009). For example, Barni et al. (2010) analyzed block-level statistics around suspected forgeries, while Chen & Hsu (2008) trained SVMs to discriminate forged from authentic regions. Other approaches modeled the probability of double compression at the DCT-block level Bianchi & Piva (2012) or extracted block artifact grids (BAGs) to localize tampering via grid discontinuities Li et al. (2009). For a comprehensive overview of classical approaches for JPEG-based forensics, see Verdoliva (2020).

Deep Learning for Forgery Detection Deep learning shifted the field toward end-to-end detectors that combine RGB and frequency-domain or additional noise features. CNN-based approaches (Bayar & Stamm, 2018; Zhou et al., 2018; Amerini et al., 2017) and hybrid two-stream models (Kwon et al., 2021; Dong et al., 2022) demonstrated strong results on natural image tampering. More recently, attention-based and transformer-based architectures (Liu et al., 2022; Wang et al., 2022a) improved global reasoning but often lose sensitivity to subtle local artifacts. However, most of these methods remain optimized only for natural images, where manipulations are larger and visually

distinct, rather than document forgeries where edits are localized and text-like (Wu et al., 2019; Nandanwar et al., 2021).

Deep Learning for Document Forgery Detection Since document forgeries are much more subtle compared to natural images, recent models explicitly introduce frequency-domain feature fusion strategies into deep neural networks for enhanced tampering detection. Abramova & Böhme (2016) proposed a method for detecting copy-move tampering in document images based on double quantization artifacts, however, this approach falls short when faced with multiple JPEG compressions. Wang et al. (2022b) introduced a two-stream Faster R-CNN (Ren et al., 2015) combining RGB and frequency features, but primarily targets SRNet-generated forgeries (Wu et al., 2019) rather than careful copy-paste tampering. For instance, Document Tampering Detector (DTD) Qu et al. (2023) is a recent state-of-the-art model a multi-modality Swin Transformer (Liu et al., 2021) model that employs a Frequency Perception Head (FPH) to capture tampering clues from DCT coefficients and a Multi-view Iterative Decoder (MID) to leverage multi-scale feature information from separte pixel-domain and frequency-domain input streams. FFDN (Chen et al., 2025) propose the Wavelet-like Frequency Enhancement (WFE) module for adaptive fusion of pixel-domain and frequency-domain features and present current state-of-the-art performance on multiple document tampering benchmarks. DocForgenet (Riaz et al., 2025) recently also propose to enchance feature fusion using dual-cross stream networks that fuse the freuqency and pixel-levle features via cross-attention.

Despite several architectural advances, most of the current SotA methods primarily rely on the block-level JPEG DCT coefficients for tampering detection. As we demonstrate in this work, this dependency creates a fundamental vulnerability in these methods. That is, by designing forgeries that preserve the local statistics of the DCT coefficients while during tampering, detectors can be catastrophically misled. Our work is the first to systematically exploit this weakness and highlight the need for more robust tampering detection methods.

3 PRELIMINARIES

3.1 JPEG COMPRESSION MODEL

The encoding process of JPEG compression can be summarized in three main steps:

1. The image is partitioned into 8×8 non-overlapping blocks, and a 2D discrete cosine transform (DCT) (Ahmed et al., 1974) is applied to each block independently to compute the DCT coefficients.
2. The resulting DCT coefficients are quantized using a quantization matrix $\mathbf{Q} \in \mathbb{N}^{8\times 8}$, the values of which are determined according to the compression quality factor $f \in [0, 100]$.
3. Finally, the quantized DCT coefficients are entropy-coded (e.g., using Huffman and run-length encoding) in a lossless manner.

Formally, given an original uncompressed image block $\mathbf{I}_{ij}$, JPEG compression followed by decompression with a quality factor f can be expressed as

$$\mathbf{I}'_{ij} \;=\; \mathrm{IDCT}(\mathcal{D}(\mathcal{Q}_f(\mathrm{DCT}(\mathbf{I}_{ij})))) + \boldsymbol{\varepsilon}, \tag{1}$$

where $\mathcal{Q}_f(\cdot)$ denotes quantization with quality-dependent matrix $\mathbf{Q}$, $\mathcal{D}(\cdot)$ denotes the corresponding dequantization, and $\boldsymbol{\varepsilon}$ accounts for rounding and truncation errors during decoding.

The above expression applies to a single 8×8 block at position (i, j). The full decompressed image is obtained by concatenating all reconstructed blocks:

$$\mathcal{C}_f(\mathbf{I}) = \bigcup_{i,j} \mathbf{I}'_{ij}. \tag{2}$$

Because quantization is performed independently across blocks, horizontal and vertical discontinuities emerge at block boundaries, commonly referred to as *block artifacts*. In image forensics, inconsistencies in block artifact grids between authentic and tampered regions provide strong cues for manipulation. Figure 2 illustrates this effect.

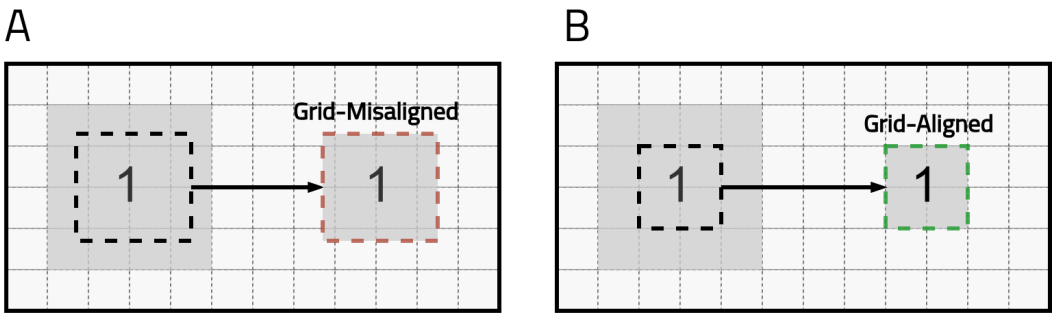


Figure 2: (a) A standard copy–move disrupts the block artifact grid, leaving forensic traces of tampering. (b) Our BAG-aware copy–move aligns the tampered text with the underlying 8×8 grid, preserving per-block DCT statistics.

4 METHODOLOGY

Let $I \in \mathbb{R}^{3\times H\times W}$ be an input document image in RGB space. Then, for a standard copy–move image tampering setup (Li et al., 2009; Qu et al., 2023; Wang et al., 2022b), let a source tampering region be defined by the bounding box $b_s = (x_s, y_s, w, h)$ and a target tampering location be defined by the bounding box $b_t = (x_t, y_t, w, h)$, where each bounding box is specified by its top-left coordinates (x, y) and width w and height h. The copy–move operator Π, which crops the source region and pastes it at the target location, can then be defined as follows:

$$\Pi(I, b_s, b_t)_{:,i,j} = \begin{cases} I_{:,\, i-y_t+y_s,\, j-x_t+x}, & x_t \le j < x_t + w,\ y_t \le i < y_t + h, \\ I_{:,i,j}, & \text{otherwise.} \end{cases} \tag{3}$$

That is, the pixels inside the target bounding box b_t are replaced by the corresponding pixels from the source bounding box b_s, while all other image pixels remain unchanged. Following prior works (Li et al., 2009; Qu et al., 2023; Wang et al., 2022b), we assume that after the copy-move operation is applied, the image again undergoes one or multiple JPEG compressions with a set of quality factors $F = \{f_1, f_2, \ldots, f_n\}$ and stored, resulting in the final tampered image I':

$$I' \;=\; (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1})(\Pi(x, b_s, b_t)) \tag{4}$$

Assuming a deep forgery detector $f_\theta : \mathbb{R}^{3\times H\times W} \to [0,1]^{H\times W}$ that outputs a tampering probability map $\hat{y} = f_\theta(I')$ for the forged image I', a generalized copy-move forgery can be modeled as constrained adversarial attack Zhou et al. (2022) that aims to minimize the detector's response over all desired tampered regions $b_t \in \mathcal{T}$:

$$\min_{b_s\in\mathcal{S},\, b_t\in\mathcal{T}} \sum_{b_t\in\mathcal{T}} \sum_{(i,j)\in\mathcal{P}(b_t)} \hat{y}_{i,j}, \tag{5}$$

where

$$\mathcal{P}(b_t) = \{(i,j) \mid x_t \le j < x_t + w,\ y_t \le i < y_t + h\}.$$

Whereas $\mathcal{S}$ and $\mathcal{T}$ denote the desired candidate sets of source and target forgery regions. Directly solving Eq. equation 5 is intractable for two reasons. First, selecting appropriate candidate bounding boxes (b_s, b_t) is nontrivial: the forger must identify semantically meaningful text regions that can be imperceptibly aligned with the target regions in RGB space, and the coordinates and sizes of these boxes can vary arbitrarily, leading to an exponentially large search space. Second, in realistic scenarios the forger does not have white-box access to the model f_θ, making direct evaluation of $\hat{y}$ infeasible. While the first difficulty can be substantially mitigated using modern OCR tools, which we also employ to define the set of source candidates $\mathcal{S}$, the second limitation persists. To circumvent this, we propose tackling the problem indirectly by exploiting structural biases of modern detectors f_θ, such as their over-reliance on frequency-domain DCT features for tampering detection.

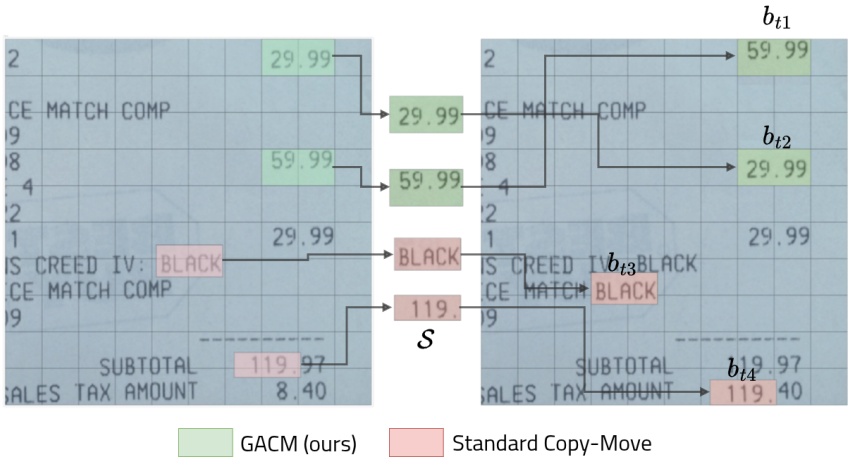


Figure 3: As shown, compared to standard copy-move setup, GACM first aligns the source text boxes $\{b_1, b_2, \ldots, b_n\}$ to the closest grid frontiers and then paste them to target tampering locations also aligned with the grid of the target locations. Each cell of the grid is of size 8×8.

Algorithm 1 Attack 1: Grid-Aligned Copy–Move (GACM)

Require: Image $I \in \mathbb{R}^{3\times H\times W}$; OCR bounding boxes $\mathcal{S}_{\text{OCR}} = \{(x_i, y_i, w_i, h_i, \text{conf}_i)\}_{i=1}^{N}$; JPEG quality factors $F = \{f_1, \ldots, f_n\}$; target tampering box b_t; bounding box confidence threshold τ_{conf}; size match threshold τ_{area}

Ensure: Forged image I', mask M

1: **fn** SNAP8$(b) := (8\lfloor x/8 \rfloor, 8\lfloor y/8 \rfloor, 8\lceil w/8 \rceil, 8\lceil h/8 \rceil)$, where $b = (x, y, w, h)$ ▷ Align box to 8-pixel grid

2: $\mathcal{S} \leftarrow \left\{ \text{SNAP8}(b) \;\middle|\; (b, \text{conf}) \in \mathcal{S}_{\text{OCR}} \;\wedge\; \text{conf} \geq \tau_{\text{conf}} \wedge \right\}$ ▷ Filter OCR boxes by confidence and overlap

3: $\bar{b}_s \leftarrow \arg\min_{\substack{\bar{b}_s \in \bar{\mathcal{S}} \\ \bar{b}_s \neq \bar{b}_t}} |\text{area}(\bar{b}_s) - \text{area}(\bar{b}_t)|$ ▷ Select source box with similar size to target

s.t. $\frac{\min(\text{area}(\bar{b}_s), \text{area}(\bar{b}_t))}{\max(\text{area}(\bar{b}_s), \text{area}(\bar{b}_t))} \geq 1 - \tau_{\text{area}},\ \text{IoU}(\bar{b}_s, \bar{b}_t) \leq \epsilon$

4: $I_{\text{cm}} \leftarrow \Pi(I, \bar{b}_s, \bar{b}_t)$ ▷ Copy–move patch from $\bar{b}_s$ to $\bar{b}_t$

5: $M \leftarrow 0;\ \Omega \leftarrow [x_t : x_t + w) \times [y_t : y_t + h);\ M(\Omega) \leftarrow 1$ ▷ Update mask for tampered region

6: $I' \leftarrow (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1})(I_{\text{cm}})$ ▷ Apply JPEG compression pipeline

7: **return** I', M

4.1 Attack 1: Grid-Aligned Copy–Move (GACM)

As described in Section 3.1, JPEG compression operates independently on 8×8 blocks, and modern forgery detectors Qu et al. (2023); Riaz et al. (2025); Chen et al. (2025); Kwon et al. (2021) exploit quantization artifacts localized within these blocks as strong cues for tampering detection. Building on this observation, we propose the Grid-Aligned Copy–Move (GACM) forgery attack, which aligns tampering with the underlying JPEG block structure to minimize detectable inconsistencies (see Figure 3). A complete pseudocode for our algorithm is defined in Algorithm 1. For OCR-detected text boxes, both source and target regions are snapped to the 8×8 JPEG grid, ensuring that pasted content coincides exactly with block boundaries. Formally, for a bounding box $b = (x, y, w, h)$, we define

$$\mathcal{A}(x, y, w, h) = \left(8\left\lfloor \tfrac{x}{8} \right\rfloor,\ 8\left\lfloor \tfrac{y}{8} \right\rfloor,\ 8\left\lceil \tfrac{w}{8} \right\rceil,\ 8\left\lceil \tfrac{h}{8} \right\rceil\right),$$

and select source and target boxes b_s and b_t of similar area with low overlap with known forgeries, setting $\bar{b}_s = \mathcal{A}(b_s)$ and $\bar{b}_t = \mathcal{A}(b_t)$. Then for a given target tampering region b_t, we search for boxes b_s in the source candidate set $\mathcal{S}$ that aligns with the target b_t in terms of size and background similarity. The source box b_s is then copied to the target box b_t by applying the copy-move operator (see Eq. 3) to perform the tampering. Finally, the manipulated image is recompressed using the same

Algorithm 2 Attack 2: Grid Shift via Pad–Recompress–Crop (PRC)

Require: Image $I \in \mathbb{R}^{3\times H\times W}$; shift policy π (fixed or random); JPEG qualities $\mathbf{q} = [q_1, \ldots, q_m]$; padding mode $\phi \in \{\text{edge}, \text{const}, \text{reflect}\}$; optional SSIM floor τ_{ssim}
Ensure: Grid-shifted image I'
1: $(\Delta x, \Delta y) \leftarrow \pi$, with $(\Delta x, \Delta y) \in \{0, \ldots, 7\}^2 \setminus \{(0, 0)\}$
2: $I_p \leftarrow$ Pad I with $(\Delta x, \Delta y)$ using mode ϕ ▷ Pad
3: $I_c \leftarrow (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1})(I_{\text{p}})$ ▷ Recompress
4: $I' \leftarrow I_c[\Delta y : \Delta y + H,\ \Delta x : \Delta x + W]$ ▷ Crop
5: **return** I'

(or divisibility-compatible) JPEG quality factors as described in Eq. 4, maintaining quantization alignment and preserving BAG consistency.

4.2 Attack 2: Grid Shift via Pad–Recompress–Crop (PRC)

While the GACM attack aims to minimize detector responses on target tampered regions, modern detectors' reliance on frequency-domain block artifacts suggests a complementary vulnerability. If these models discriminate forged from unaltered regions based on slight misalignments in the JPEG block grid, then deliberately introducing small global grid distortions could trigger the detector to classify many pixels as manipulated. Intuitively, this can be viewed as another type of adversarial attack that solves the inverse problem to Eq. 5: rather than minimizing detector responses, we seek to maximize the predicted tampering probability across the entire image. Formally, let $\Delta x, \Delta y$ define the grid shifts in horizontal and vertical directions, respectively, then Pad–Recompress–Crop (Π_{PRC}) operator for grid shift is defined as follows:

$$\Pi_{PRC}(I, \Delta x, \Delta y) = R_{\Delta x,\Delta y} \circ \mathcal{C}_q \circ P_{\Delta x,\Delta y}(I),$$

where $P_{\Delta x,\Delta y}$ pads the image on the left and top by $(\Delta x, \Delta y)$ pixels, and $R_{\Delta x,\Delta y}$ crops these pixels after JPEG recompression step described in Eq. 4. Applying this operator produces the attacked image

$$I' = \Pi_{PRC}(I, \Delta x, \Delta y) \tag{6}$$

The PRC attack is then formulated as an optimization over the grid shift $(\Delta x, \Delta y)$:

$$\max_{\Delta x,\Delta y} \sum_{(i,j)\in I'} f_\theta(I')_{i,j}, \quad \text{s.t. } (\Delta x, \Delta y) \neq (0, 0),\ 0 \leq \Delta x \leq 7,\ 0 \leq \Delta y \leq 7. \tag{7}$$

By carefully selecting $(\Delta x, \Delta y)$, the Pad–Recompress–Crop (PRC) attack aims to exploits the model's sensitivity to grid misalignment, with the goal of producing as many false-positive tampering predictions as possible. Full pseudocode for the PRC attack is provided in Algorithm 2.

5 Experiments

5.1 Experimental Setup

Datasets For all experimental evaluation, we use **DocTamper** (Qu et al., 2023), the largest publicly available dataset for document tampering detection. DocTamper contains a total of 170k tampered document images in English and Chinese languages, with tampering done on the dataset using various methods such as copy-move, splicing, and generation. For evaluation, it provides a training set of 120k samples, a primary testing set D-TestingSet with 30k samples, and two cross-domain testing sets DocTamper-FCD with 2k samples and DocTamper-SCD with 18k samples. All images in the dataset are pre-forgered and stored uncompressed, with pixel-level annotations of tampered text regions provided as ground-truth masks. It is worth mentioning that the DocTamper-FCD split is derived from the Noisy Office Dataset (Castro-Bleda et al., 2019), while DocTamper-SCD split comes from HUAWEI Cloud dataset (Huawei Cloud, 2022), and therefore even within the testing splits there is a diverse sample distribution.

Models. We evaluate our proposed forgery attacks on three state-of-the-art forgery detectors that all rely on *frequency-domain DCT features* as one of their primary cues for localizing the tampered regions. Specifically, we consider (a) DTD Qu et al. (2023), (b) DocForgeNet Riaz et al. (2025), and (c) FFDN Chen et al. (2025). These methods represent the current best-performing approaches on DocTamper and exemplify the block-artifact–guided family of detectors. For all our experiments, we directly use publicly released training checkpoints for each of these models.

Evaluation Protocol. We report pixel-wise Precision (P), Recall (R), and F1-score (F) on the DocTamper benchmark using its three standard splits: D-TestingSet, DocTamper-FCD, and DocTamper-SCD. Following Qu et al. (2023), test images undergo 1–3 JPEG recompressions with quality factors ≥ 75 using the public seed. We compare the original DocTamper tampering against our Grid-Aligned Copy–Move (GACM) retampering that snaps source/target boxes to the JPEG 8×8 grid prior to the same recompression schedule. Since DocTamper does not provide original images and only the forged ones, we re-tamper these existing forged images using GACM and report the final evaluation metrics for GACM without considering the initially forged areas.

Implementation Details. To evaluate our proposed GACM attack on the complete dataset, we obtain the source ($\mathcal{S}$) and target ($\mathcal{T}$) boxes for all dataset splits using the Tesseract OCR (Kay, 2007), and randomly select target and source boxes (excluding those already forged in DocTamper (Qu et al., 2023)) based on the similarity of their sizes with a given threshold. We also ensure background consistency between target and source boxes using a similar threshold. This is done to maintain legibility in RGB space and to allow full exploitation of the DCT streams by the models. With this protocol, we generate grid-aligned tamperings on all three testing splits defined in the DocTamper dataset, resulting in a new attack benchmark dataset for consistent evaluation of our attacks across different models.

5.2 Quantitative Evaluation: GACM and PRC

Table 1 shows the results of our proposed document forgery attacks GACM and PRC, where we compare the performance of different document forgery detectors on original DocTamper tamperings vs the image tampering done using our attacks. The evaluation metrics are computed as described in Section 5.1. As evident from the results, the detection performance of these methods is remarkable under the standard protocol but drops significantly in case of our proposed GACM and PRC attacks across all the test splits. The performance drop on the the Doctamper-FCD test split is particularly noticeable, with both DTD Qu et al. (2023) and DocForgeNet Riaz et al. (2025) models failing catastrophically on both attacks, with F1-scores dropping to as low as 3.1%. FFDN , on the other hand, showed considerably higher robustness against our tampering attacks compared to its coutnerparts, which suggests that it may have learned better representations. However, it is worth mentioning that FFDN additionally employs a Visual Enhancement Module (VFM) module which adaptively fuses the RGB and frequency-domain features in an attempt to reduce its reliance on the frequency-domain features and therefore its imrpoved robustness is justified. However, it still

Table 1: Quantitative comparison of the GACM forgery attack on the three testing splits of DocTamper dataset (Qu et al., 2023). **P**, **R**, and **F** denote the pixel-wise precision, recall, F1-scores. Our proposed GACM forgery attack effectively bypasses detection by existing state-of-the-art DL-based document forgery detection methods as evident by the considerable drop in F1 scores.

Method	Forgery Method	D-TestingSet			DocTamper-FCD			DocTamper-SCD		
		P	R	F	P	R	F	P	R	F
DTD (Qu et al., 2023)	DocTamper Tampering	0.752	0.701	0.726	0.783	0.742	0.762	0.698	0.701	0.700
	GACM (ours)	0.410	0.167	0.237	0.140	0.017	0.031	0.710	0.289	0.411
	PRC (ours)	0.266	0.942	0.314	0.003	0.195	0.007	0.201	0.503	0.225
DocForgeNet (Riaz et al., 2025)	DocTamper Tampering	0.802	0.751	0.774	0.845	0.801	0.822	0.701	0.739	0.720
	GACM (ours)	0.400	0.185	0.253	0.160	0.024	0.042	0.776	0.356	0.488
	PRC (ours)	0.129	0.852	0.205	0.292	0.471	0.065	0.252	0.476	0.248
FFDN (Chen et al., 2025)	DocTamper Tampering	0.960	0.928	0.944	0.948	0.921	0.934	0.859	0.851	0.856
	GACM (ours)	0.811	0.459	0.586	0.689	0.295	0.406	0.800	0.615	0.696
	PRC (ours)	0.851	0.902	0.805	0.922	0.511	0.658	0.831	0.639	0.722

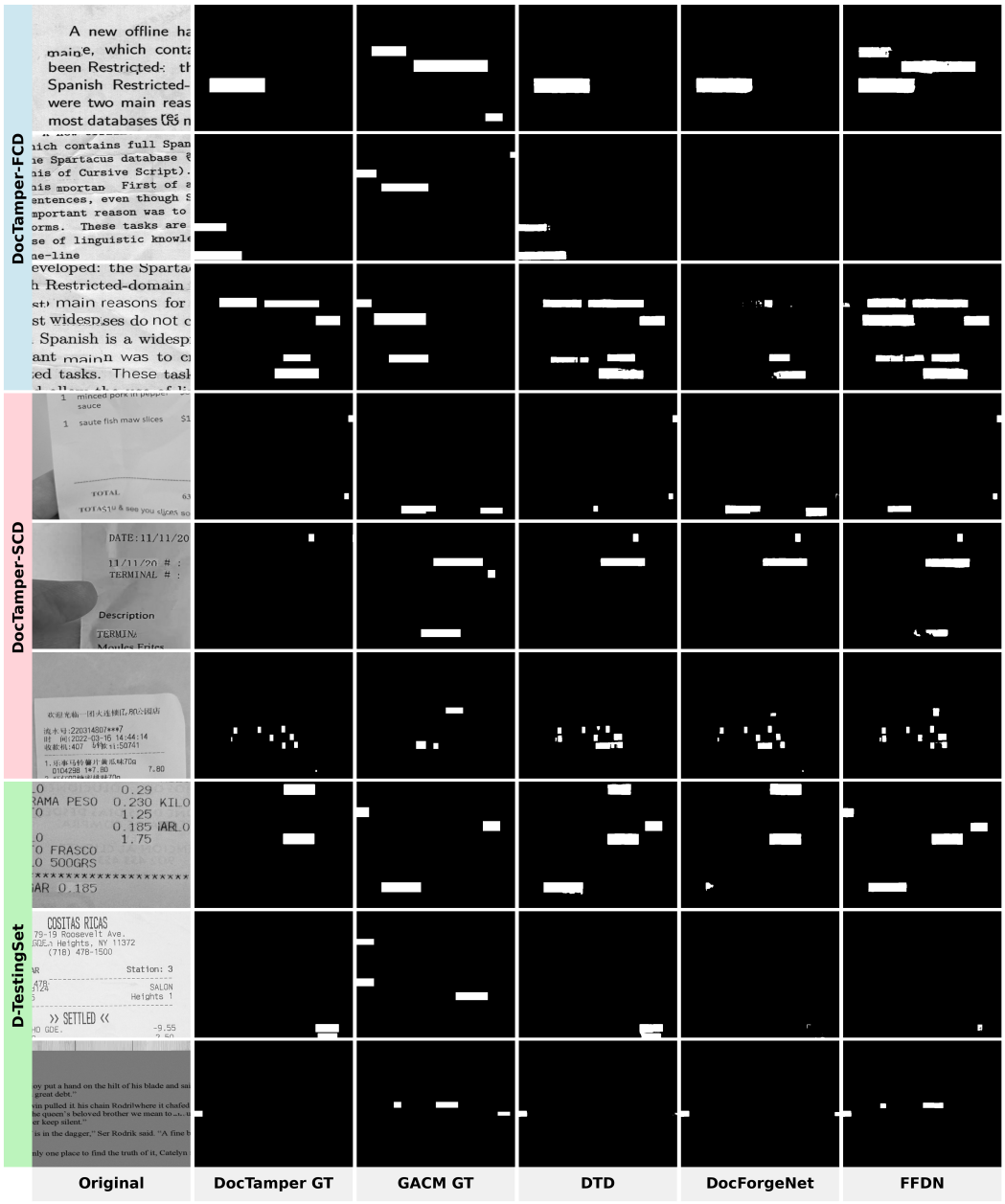


Figure 4: Qualitative comparison of the GACM forgery attack across different state-of-the-art detection methods on the DocTamper dataset (Qu et al., 2023).

performs considerably worse in comparison to its baseline performance on the standard tampering setup which highlights the effectiveness of our attacks even in case of adaptive fusion. Overall, the pattern supports our hypothesis that existing DL-based forgery detection models may have been relying on the local validity of the block-level DCT features instead of learning more global semantic representations of tampering.

5.3 Qualitative Analysis: GACM and PRC

Fig 4 shows the qualitative results of the GACM tampering attack, where we compare the performance of different document forgery detectors on randomly selected samples from the three test splits of the DocTamper dataset Qu et al. (2023). We see the effectiveness of our grid-aligned re-

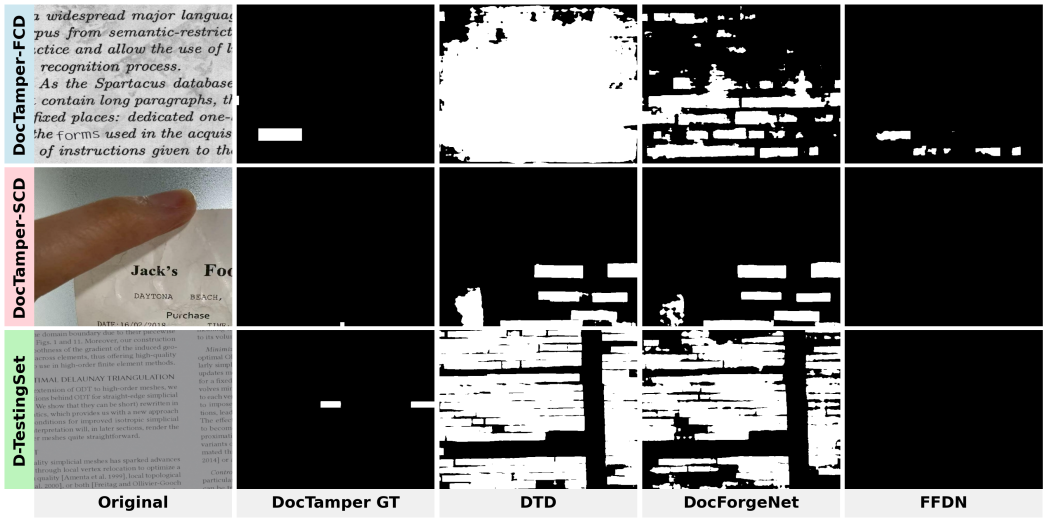


Figure 5: Qualitative comparison of the PRC forgery attack across different state-of-the-art detection methods on the DocTamper dataset (Qu et al., 2023).

tampering approach where state-of-the-art models like DTD and DocForgeNet show a catastrophic failure to detect any such forgeries even thou they are visually easy to detect. This shows there inherent dependency on shortcut representation learning relying on DCT grid-artifact signal for tampering detection. Such failures are also visible for FFDN but far less than the other two models as FFDN architecture use wavelet like feature enhancement and do not entirely depend on DCT signal for the tampered region detection.

Fig 5 shows a different approach to our attack. Knowing the models reliance on DCT signal for grid-artifacts inconsistencies we can exploit them by manipulating the grid. Via PRC we show that grid shift causes the models to foolishly trigger multiple tamperings even thou the document is not tampered in those regions at all. This result is again much more visible on state-of-the-art models that have heavy reliance on DCT features whereas for FFDN it still helps in diminishing the model's localizations.

6 Conclusions

We have introduced two novel adversarial forgery attacks that exploit the over-reliance of state-of-the-art document forgery detectors on frequency-domain DCT features. Our experiments demonstrate that with minor grid-manipulation, existing SotA document tampering detection methods can be catastrophically fooled, with detection rates reduced to near-random levels, and that grid manipulations can systematically trigger false positives, leaving these models unreliable. In future, it could be interesting to investigate the applicability of our attacks on natural image forgery detectors.

7 Broader Impact

Our work highlights critical safety and reliability concerns in current state-of-the-art document forgery detection systems, showing that over-reliance on block-level JPEG artifacts can be exploited to bypass automated safeguards. These vulnerabilities have direct implications for security and fairness, as malicious actors could selectively manipulate documents or trigger false positives, potentially affecting individuals or organizations disproportionately. By exposing these weaknesses, our research encourages the development of more robust, alignment-conscious systems that prioritize semantically grounded representations, enhancing societal trust in AI-mediated document verification. In addition, our proposed forgery attacks can serve as a new form of evaluation benchmark for future research to audit the overall robustness of forgery detection models.

REFERENCES

Svetlana Abramova and Rainer Böhme. Detecting copy-move forgeries in scanned text documents. In *Media Watermarking, Security, and Forensics*, 2016. URL https://api.semanticscholar.org/CorpusID:4468868.

N. Ahmed, T. Natarajan, and K.R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, 1974. doi: 10.1109/T-C.1974.223784.

Irene Amerini, Tiberio Uricchio, Lamberto Ballan, and Roberto Caldelli. Localization of jpeg double compression through multi-domain convolutional neural networks, 2017. URL https://arxiv.org/abs/1706.01788.

M. Barni, A. Costanzo, and L. Sabatini. Identification of cut and paste tampering by means of double-jpeg detection and image segmentation. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pp. 1687–1690, 2010. doi: 10.1109/ISCAS.2010.5537505.

Belhassen Bayar and Matthew C. Stamm. Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection. *IEEE Transactions on Information Forensics and Security*, 13(11):2691–2706, 2018.

Tiziano Bianchi and Alessandro Piva. Image forgery localization via block-grained analysis of jpeg artifacts. *IEEE Transactions on Information Forensics and Security*, 7(3):1003–1017, 2012. doi: 10.1109/TIFS.2012.2187516.

M J Castro-Bleda, S España-Boquera, J Pastor-Pellicer, and F Zamora-Martínez. The noisyoffice database: A corpus to train supervised machine learning filters for image processing. *The Computer Journal*, 63(11):1658–1667, 11 2019. ISSN 0010-4620. doi: 10.1093/comjnl/bxz098. URL https://doi.org/10.1093/comjnl/bxz098.

Yi-Lei Chen and Chiou-Ting Hsu. Image tampering detection by blocking periodicity analysis in jpeg compressed images. pp. 803–808, 10 2008. doi: 10.1109/MMSP.2008.4665184.

Zhongxi Chen, Shen Chen, Taiping Yao, Ke Sun, Shouhong Ding, Xianming Lin, Liujuan Cao, and Rongrong Ji. Enhancing tampered text detection through frequency feature fusion and decomposition. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 200–217, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73414-4.

Chengbo Dong, Xinru Chen, Ruohan Hu, Juan Cao, and Xirong Li. Mvss-net: Multi-view multi-scale supervised networks for image manipulation detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP, 06 2022. doi: 10.1109/TPAMI.2022.3180556.

Zhigang Fan and R.L. de Queiroz. Identification of bitmap compression history: Jpeg detection and quantizer estimation. *IEEE Transactions on Image Processing*, 12(2):230–235, 2003. doi: 10.1109/TIP.2002.807361.

Huawei Cloud. Huawei cloud visual information extraction competition. https://www.huaweicloud.com/, 2022. Accessed: 2025-09-25.

Anthony Kay. Tesseract: an open-source optical character recognition engine. *Linux J.*, 2007(159): 2, July 2007. ISSN 1075-3583.

Myung-Joon Kwon, In-Jae Yu, Seung-Hun Nam, and Heung-Kyu Lee. Cat-net: Compression artifact tracing network for detection and localization of image splicing. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 375–384, 2021.

Weihai Li, Yuan Yuan, and Nenghai Yu. Passive detection of doctored jpeg image via block artifact grid extraction. *Signal Processing*, 89(9):1821–1829, 2009. ISSN 0165-1684. doi: https://doi.org/10.1016/j.sigpro.2009.03.025. URL https://www.sciencedirect.com/science/article/pii/S0165168409001315.

Xiaohong Liu, Yaojie Liu, Jun Chen, and Xiaoming Liu. Pscc-net: Progressive spatio-channel correlation network for image manipulation detection and localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32:1–1, 11 2022. doi: 10.1109/TCSVT.2022.3189545.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer V2: scaling up capacity and resolution. *CoRR*, abs/2111.09883, 2021. URL https://arxiv.org/abs/2111.09883.

Lokesh Nandanwar, Palaiahnakote Shivakumara, Umapada Pal, Tong Lu, Daniel Lopresti, Bhagesh Seraogi, and Biswadeep Chaudhuri. A new method for detecting altered text in document images. *International Journal of Pattern Recognition and Artificial Intelligence*, 35:2160010, 09 2021. doi: 10.1142/S0218001421600107.

Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, and Rafael Grompone von Gioi. Local JPEG Grid Detector via Blocking Artifacts, a Forgery Detection Tool. *Image Processing On Line*, 10: 24–42, 2020. https://doi.org/10.5201/ipol.2020.283.

Abhinandan Kumar Pun, Mohammed Javed, and David S. Doermann. A survey on change detection techniques in document images, 2023. URL https://arxiv.org/abs/2307.07691.

Chenfan Qu, Chongyu Liu, Yuliang Liu, Xinhong Chen, Dezhi Peng, Fengjun Guo, and Lianwen Jin. Towards robust tampered text detection in document image: New dataset and new solution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5937–5946, June 2023.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015. URL http://arxiv.org/abs/1506.01497.

Nauman Riaz, Stefan Agne, Andreas Dengel, and Sheraz Ahmed. Docforgenet: Dual cross-stream fusion network for robust forgery detection in scanned documents. In Xu-Cheng Yin, Dimosthenis Karatzas, and Daniel Lopresti (eds.), *Document Analysis and Recognition – ICDAR 2025*, pp. 329–346, Cham, 2025. Springer Nature Switzerland.

Luisa Verdoliva. Media forensics and deepfakes: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932, 2020. doi: 10.1109/JSTSP.2020.3002101.

Junke Wang, Zuxuan Wu, Jingjing Chen, Xintong Han, Abhinav Shrivastava, Ser-Nam Lim, and Yu-Gang Jiang. Objectformer for image manipulation detection and localization, 2022a. URL https://arxiv.org/abs/2203.14681.

Qing Wang and Rong Zhang. Double JPEG compression forensics based on a convolutional neural network. *EURASIP J. Multimed. Inf. Secur.*, 2016(1), December 2016.

Yuxin Wang, Boqiang Zhang, Hongtao Xie, and Yongdong Zhang. Tampered text detection via RGB and frequency relationship modeling. *Chinese Journal of Network and Information Security*, 8(3):29, 2022b. doi: 10.11959/j.issn.2096-109x.2022035. URL https://www.infocomm-journal.com/cjnis/EN/abstract/article_172502.shtml.

Liang Wu, Chengquan Zhang, Jiaming Liu, Junyu Han, Jingtuo Liu, Errui Ding, and Xiang Bai. Editing text in the wild. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1500–1508, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3343031.3350929. URL https://doi.org/10.1145/3343031.3350929.

Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Learning rich features for image manipulation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Shuai Zhou, Chi Liu, Dayong Ye, Tianqing Zhu, Wanlei Zhou, and Philip S. Yu. Adversarial attacks and defenses in deep learning: From a perspective of cybersecurity. *ACM Comput. Surv.*, 55(8), December 2022. ISSN 0360-0300. doi: 10.1145/3547330. URL https://doi.org/10.1145/3547330.