Mistaking Periodicity for Manipulation: JPEG-Induced Structural Bias in Document Tampering Detection

Nauman Riaz[*1,2,3]
nauman.riaz@dfki.de
Ali Hussain[*4]
ahussain.bscs23seecs@seecs.edu.pk
Saifullah Saifullah[*1,2,3]
saifullah.saifullah@dfki.de
Muhammad Imran Malik[3,4]
malik.imran@seecs.edu.pk
Stefan Agne[2,3]
stefan.agne@dfki.de
Andreas Dengel[2]
andreas.dengel@dfki.de
Sheraz Ahmed[2,3]
sheraz.ahmed@dfki.de

[1] Department of Computer Science
RPTU Kaiserslautern-Landau
Kaiserslautern, Germany

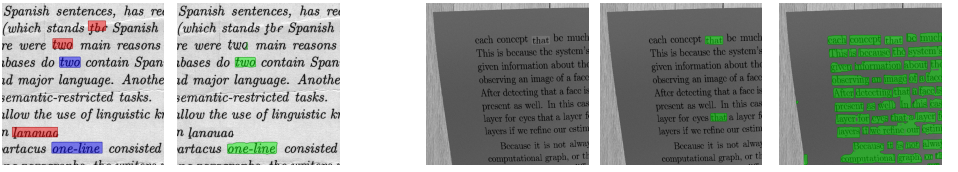
[2] Smart Data and Knowledge Services
German Research Center for Artificial Intelligence (DFKI)
Kaiserslautern, Germany

[3] DeepReader GmbH
Kaiserslautern, Germany

[4] School of Electrical Engineering and Computer Science (SEECS)
National University of Sciences and Technology (NUST)
Islamabad, Pakistan

Abstract

Deep models for document tampering detection increasingly rely on multimodal RGB+DCT architectures, implicitly assuming that JPEG block artifact grids (BAGs) provide stable cross-forgery cues. In this paper, we show that this assumption embeds a strong inductive bias that fails under minimal, adversarially constructed perturbations. Unlike natural images, where JPEG alignment is largely stochastic, document images contain sharply bounded glyph structures, making grid-aligned manipulations trivial for an adversary. We formalize this phenomenon through two complementary attacks. Grid-Aligned Forgery (GAF) preserves local JPEG block statistics by aligning copy–move, splicing, or generative manipulations to the underlying 8×8 grid, suppressing the inconsistencies on which current models depend. Pad–Recompress–Crop (PRC) globally shifts the JPEG grid while preserving semantic content and image geometry. Together, they probe whether detectors meaningfully fuse RGB and DCT features or instead memorize position-dependent frequency cues. To quantify these effects, we use two evaluation metrics, Detection Failure Rate (DFR) for missing forged regions and False Positive Area (FPA) for unintended detections, which capture failure modes not measured by prior work. Evaluations on the DocTamper and TSROIE benchmarks show that both attacks substantially degrade performance across a range of state-of-the-art and robustness-oriented detectors, including adversarially trained models, such as CAT-Net, DTD, FFDN, DocForgeNet, and ADCD-Net. Our findings indicate that many existing models exhibit a strong bias toward JPEG-grid statistics, highlighting the need for more robust multimodal architectures for real-world, security-critical document forensics.



(a) GAF attack (b) PRC attack

Figure 1: Examples showing the effectiveness of our proposed forgery attacks. (a) The Grid-Aligned Forgery (GAF) attack (red) preserves local JPEG block statistics and successfully evades detection (green), in contrast to standard forgeries (blue). (b) The Pad–Recompress–Crop (PRC) attack (right) triggers a considerable amount of false positives by introducing only a subtle grid misalignment, as seen by the model's predictions before (middle) and after the shift (right).

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 [25]. Meanwhile, the rapid progress and widespread availability of modern image editing technologies have made it increasingly convenient to create such forgeries, necessitating the development of efficient and robust methods for forgery detection [19, 20, 30].

Recent deep learning (DL)-based detectors [9, 21, 23, 28] have demonstrated strong performance on standard document tampering benchmarks [21, 28]. However, many of these methods still largely rely on exploiting the frequency-domain artifact traces introduced by JPEG compression, particularly the discontinuities in the block artifact grids [15], as discriminative cues for detecting manipulated regions in the image. While this 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: instead of learning robust, semantically meaningful evidence of tampering, models may overfit to local grid statistics and fail whenever these statistics are preserved or systematically shifted. This reflects a problematic inductive bias driven by the training data distribution and by the structure of JPEG compression, which operates on non-overlapping 8×8 image patches.

To examine this failure mode, we introduce two complementary adversarial forgery procedures. (1) Grid-Aligned Forgery (GAF) aligns manipulated regions to the JPEG grid, preserving block-level DCT statistics. Despite the manipulated content remaining visually obvious to humans, grid preservation is often sufficient to bypass existing detectors. GAF generalizes across copy–move, splicing, and generative forgeries. (2) Pad–Recompress–Crop (PRC) shifts the JPEG grid through a fixed, query-free sequence of padding, recompression, and cropping. The resulting half-block shift preserves semantic content and image geometry while changing the position of the compression grid. PRC therefore probes whether detectors have learned meaningful cross-modal correlations or have instead memo-

*These authors contributed equally to this work.

rized block-level statistics. In principle, a model with robust global representations should remain stable under such a shift; empirically, we show that this is often not the case.

We evaluate our proposed methods on DocTamper [21], one of the largest available document tampering benchmarks, as well as TSROIE [28], a receipt-based tampered text detection dataset, and compare them against several state-of-the-art deep learning–based document-tampering detection methods. Across the DocTamper benchmark, both GAF and PRC induce significant failures in state-of-the-art detectors, reducing detection rates to as low as **1%** and enabling systematic false positives. Similar degradation trends are also observed on the TSROIE evaluation, indicating that the vulnerability generalizes beyond DocTamper to other document forensics benchmarks. These results highlight a fundamental weakness in current architectures stemming from an over-reliance on JPEG-grid statistics rather than learning robust, semantically grounded tampering cues.

The main contributions of this work are the following:

- We identify and formalize a critical inductive bias in JPEG-based forgery detectors arising from block-level DCT features.
- We introduce two complementary adversarial procedures, GAF and PRC, that exploit this bias through grid preservation and controlled grid shifts.
- We show that even minimal manipulations dramatically degrade state-of-the-art detectors, revealing fundamental limitations in current multimodal JPEG–RGB document tampering detection architectures.

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 [11, 27] and were later extended to tampering localization [4, 8, 15]. For example, Barni et al. [4] analyzed block-level statistics around suspected forgeries, while Chen and Hsu [8] trained SVMs to discriminate forged from authentic regions. Other approaches modeled the probability of double compression at the DCT-block level [6] or extracted block artifact grids (BAGs) to localize tampering via grid discontinuities [15]. For a comprehensive overview of classical approaches for JPEG-based forensics, see [25].

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 [3, 5, 31] and hybrid two-stream models [10, 14] demonstrated strong results on natural image tampering. More recently, attention-based and transformer-based architectures [16, 26] 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 [19, 30].

Deep Learning for Document Forgery Detection. Since document forgeries are much subtler than natural image manipulations, recent detectors explicitly fuse pixel-domain and frequency-domain features. Abramova and Böhme [1] proposed copy–move detection based on double quantization artifacts, but this approach falls short under multiple JPEG compressions. Wang et al. [28] introduced a two-stream Faster R-CNN [22] combining RGB and

frequency features, but primarily targets SRNet-generated forgeries [30] rather than careful copy-paste tampering. Document Tampering Detector (DTD) [21] is a recent multi-modality Swin Transformer [17] model that uses a Frequency Perception Head (FPH) for DCT-based tampering cues and a Multi-view Iterative Decoder (MID) for multi-scale fusion across pixel-domain and frequency-domain streams. FFDN [9] builds on DTD by adding a Vision Enhancement Module (VEM) and a Wavelet-like Frequency Enhancement (WFE) module for adaptive pixel/frequency fusion, achieving state-of-the-art performance on multiple document tampering benchmarks. DocForgeNet [23] similarly enhances feature fusion with dual-cross stream networks that combine frequency and pixel-level features through cross-attention. In addition, recent document-oriented models further extend multi-stream fusion. ADCD-Net [29] introduces an adaptive DCT weighting mechanism to handle block misalignment and employs hierarchical content disentanglement to reduce strong text–background bias, improving robustness under resizing and recompression. Similarly, the RTM baseline ASC-Former [18] leverages consistency-aware aggregation and gated cross-neighborhood attention to fuse RGB and transformed-domain cues, demonstrating strong performance on manually edited, highly concealed forgeries. Shao et al. [24] recently studied adversarial robustness for document tampering localization, showing that representative DTL models, including the visual-frequency VF-DTL baseline corresponding to DTD [21], are highly vulnerable to gradient-based white-box and transfer-based black-box perturbations. They further proposed latent manifold adversarial training to improve robustness against such perturbations. Our work is complementary: rather than optimizing small pixel-level perturbations with model access, we expose a structural vulnerability caused by JPEG grid statistics, using grid-aligned forgeries and global JPEG-grid shifts that directly target the RGB+DCT assumptions of document detectors.

Despite several architectural advances, many current state-of-the-art document forgery detection models remain fundamentally dependent on block-level JPEG grid artifacts for identifying tampering. While this approach proves effective for natural image forgery, where random operations such as copy-move have only a 1/64 random chance of aligning with 8×8 JPEG block boundaries and thereby introduce detectable grid artifacts, document images present fundamentally different characteristics. In particular, the presence of discrete glyph structures with sharp foreground-background transitions makes it feasible to execute grid-aligned tamperings while maintaining visual plausibility. Our work is the first to systematically exploit this JPEG-grid domain-specific vulnerability, demonstrating that current detection paradigms can be reliably circumvented and underscoring the critical need for more robust document tampering detection frameworks.

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) [2] 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

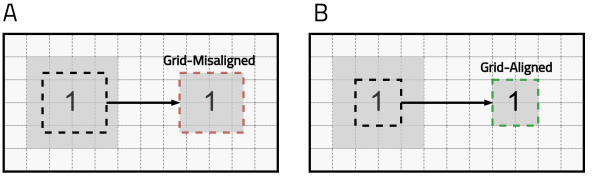


Figure 2: (a) A standard forgery disrupts the block artifact grid, leaving forensic traces of tampering. (b) Our proposed Grid-Aligned Forgery aligns the tampered text with the underlying 8×8 grid, in an attempt to preserve the per-block DCT statistics.

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

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

where $\mathcal{Q}_f(\cdot)$ denotes quantization with $\mathbf{Q}$, $\mathcal{D}(\cdot)$ denotes the corresponding dequantization, and ε accounts for rounding and truncation errors during decoding. In standard JPEG compression, Eq. 1 is applied to a single 8×8 block at position (i, j) in the image and the full decompressed image is obtained by concatenating all independently reconstructed blocks of the image:

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

Since quantization is performed independently across each 8×8 block, horizontal and vertical discontinuities emerge at block boundaries, commonly referred to as block artifacts. In image forensics, the inconsistencies in block artifact grids between authentic and tampered regions provide strong cues for manipulation as illustrated in Fig. 2.

4 Methodology

Let $I_t \in \mathbb{R}^{3\times H\times W}$ denote an input document image in RGB space. Then, for a standard image tampering setup [15, 21, 28], let I_s be a source image from which the tampered content is obtained, together with a source bounding box $b_s = (x_s, y_s, w, h)$ specifying the position and size of the region to be copied. Let a corresponding target bounding box be $b_t = (x_t, y_t, w, h)$ specifying where this content is placed within the target image I_t. Then, let Π be a unified forgery operator that crops the source region b_s from I_s and pastes it into the target region of I_t:

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

We consider three common types of forgeries in this work. For all types, the operator Π remains identical; the only difference lies in how the source image I_s is defined: **(1) Copy–move:** $I_s = I_t$, i.e., the source is the target image itself. **(2) Splicing:** $I_s \neq I_t$, i.e., the source is a different image from which the tampered region is extracted. **(3) Generative:** I_s is produced by generative or rendering approaches. Following previous works [15, 21, 28], we assume that after the forgery operation Π 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,

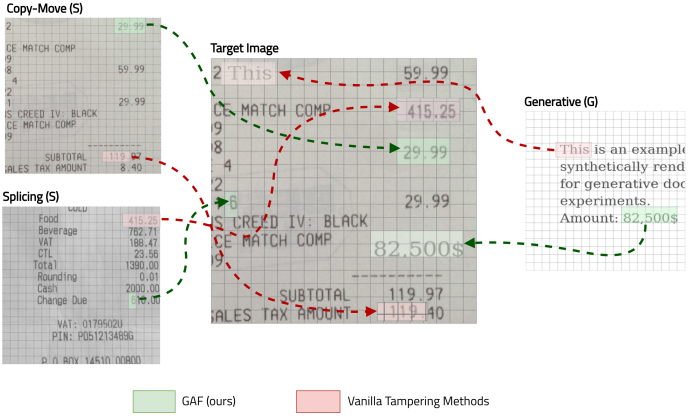


Figure 3: As shown, compared to standard tampering setups, GAF first aligns the source text boxes to the closest grid frontiers and then pastes them to target tampering locations also aligned with the grid of the target locations. Each cell of the grid corresponds to a size of 8×8.

resulting in the final tampered image I':

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

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 $\hat{y} = f_\theta(I')$ over the forged image I', the tampering operation can be modeled as a constrained adversarial attack [32] that aims to minimize the detector's response over the 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

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

and $\mathcal{S}$ and $\mathcal{T}$ denote the desired candidate sets of source and target forgery regions, respectively. However, solving this optimization problem directly 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 b_s from the source image I_s that can be imperceptibly aligned with the target regions b_t in RGB space. Since the coordinates and dimensions of these boxes can vary arbitrarily, this results in an exponentially large search space. Second, in realistic scenarios, the forger lacks white-box access to the detector 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.

4.1 Attack 1: Grid-Aligned Forgery (GAF)

JPEG compression operates independently on non-overlapping 8×8 blocks, and modern tampering detectors [9, 14, 21, 23] rely heavily on inconsistencies in block-level quanti-

Algorithm 1 Attack 1: Grid-Aligned Forgery (GAF)

Require: Target Image $I_t \in \mathbb{R}^{3\times H\times W}$; Source Image $I_s \in \mathbb{R}^{3\times H\times W}$; 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

fn $\textsc{Snap8}(b) := (8\lfloor x/8\rfloor, 8\lfloor y/8\rfloor, 8\lceil (x+w)/8\rceil - 8\lfloor x/8\rfloor, 8\lceil (y+h)/8\rceil - 8\lfloor y/8\rfloor)$ ▷ Smallest enclosing grid-aligned box

$\mathcal{S} \leftarrow \left\{ \textsc{Snap8}(b) \;\middle|\; (b, \text{conf}) \in \mathcal{S}_{\text{OCR}} \wedge \text{conf} \geq \tau_{\text{conf}} \right\}$ ▷ Filter OCR boxes by confidence and overlap

$\bar{b}_t \leftarrow \textsc{Snap8}(b_t)$

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

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

$I' \leftarrow (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1})(\Pi(I_t, I_s, \bar{b}_s, \bar{b}_t))$ ▷ Copy–move patch from $\bar{b}_s$ to $\bar{b}_t$

$M \leftarrow \mathbf{0}_{H\times W}$; $M_{i,j} \leftarrow 1$ for $(i,j) \in \mathcal{P}(\bar{b}_t)$ ▷ Update tamper mask

return I', M

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

Require: Image $I \in \mathbb{R}^{3\times H\times W}$; JPEG qualities $F = \{f_1, \ldots, f_n\}$; padding mode ϕ = replicate

Ensure: Grid-shifted image I'

$(\Delta x, \Delta y) \leftarrow (4, 4)$ ▷ Fixed query-free half-block shift

$I_p \leftarrow$ Pad I with $(\Delta x, \Delta y)$ using mode ϕ ▷ Pad

$I_c \leftarrow (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1})(I_{\text{p}})$ ▷ Recompress

$I' \leftarrow I_c[\Delta y : \Delta y + H,\ \Delta x : \Delta x + W]$ ▷ Crop; no subsequent recompression

return I'

zation artifacts as cues for manipulation. Building on this observation, we introduce Grid-Aligned Forgery (GAF), an adversarial forgery procedure that aligns manipulated regions exactly with the JPEG block structure to minimize detectable quantization mismatches (see Fig. 3), with the complete pseudocode shown in Algorithm 1. For each OCR-detected box $b = (x, y, w, h)$, we align it to the JPEG grid using

$$\textsc{Snap8}(b) = \left(x^-, y^-, x^+ - x^-, y^+ - y^-\right),$$

where $x^- = 8\lfloor x/8\rfloor$, $y^- = 8\lfloor y/8\rfloor$, $x^+ = 8\lceil (x+w)/8\rceil$, and $y^+ = 8\lceil (y+h)/8\rceil$. Thus, SNAP8 returns the smallest enclosing grid-aligned box, moving each boundary outward by fewer than eight pixels. Only OCR boxes above τ_{conf} are retained. Given an aligned target $\bar{b}_t$, we select the source $\bar{b}_s$ whose area is closest to that of $\bar{b}_t$, subject to the area-ratio and low-overlap constraints in Algorithm 1. The operator Π then copies the selected content, the mask records the aligned target region, and the forged image undergoes the same JPEG recompression schedule as its paired control.

Algorithm 1 shows GAF-CM, where source and target content come from the same document. GAF-S uses a source from another document, while GAF-G inserts rendered text; in both cases the source/target construction changes but SNAP8 and the recompression principle remain the same. The alignment specifically targets boundary-level JPEG-grid consistency. Appendix A provides implementation details for GAF-S and GAF-G.

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

While GAF aims to suppress detector responses within forged regions, the reliance of modern detectors on frequency-domain block artifacts suggests a complementary vulnerability. If these models distinguish forged from authentic regions through local misalignments in the JPEG grid, then shifting that grid globally may cause authentic content to be classified as manipulated. PRC is designed to expose this failure mode by changing which pixels share an 8×8 DCT block while preserving the document's semantic content and spatial layout.

For horizontal and vertical shifts $(\Delta x, \Delta y)$, the Pad–Recompress–Crop operator is defined as

$$\Pi_{PRC}(I, \Delta x, \Delta y) = R_{\Delta x,\Delta y} \circ (\mathcal{C}_{f_n} \circ \cdots \circ \mathcal{C}_{f_1}) \circ P_{\Delta x,\Delta y}(I),$$

where $P_{\Delta x,\Delta y}$ pads the image on the left and top, the sequence of compression operators follows Eq. 4, and $R_{\Delta x,\Delta y}$ crops the result back to its original dimensions. This construction restores the original pixel coordinates after recompression but leaves the JPEG block grid shifted. An idealized white-box attacker could then formulate the worst-case objective

$$\max_{\Delta x,\Delta y} \sum_{(i,j)\in\Omega} f_\theta(\Pi_{PRC}(I, \Delta x, \Delta y))_{i,j}, \quad \text{s.t. } (\Delta x, \Delta y) \neq (0,0),\ 0 \leq \Delta x \leq 7,\ 0 \leq \Delta y \leq 7. \tag{6}$$

This objective motivates PRC but is not optimized in our experiments. Every reported result instead uses the same deterministic half-block shift, $\Delta x = \Delta y = 4$, selected independently of the image, detector output, model, and dataset split (Algorithm 2). This query-free protocol provides a controlled and reproducible diagnostic of grid sensitivity.

5 Experiments

5.1 Experimental Setup

Datasets. We perform all evaluations on **DocTamper** [21], the largest publicly available dataset for document tampering detection. DocTamper provides 170k tampered English and Chinese document images created using copy–move, splicing, and generative methods. The dataset includes 120k training samples, a 30k primary test split (D-TestingSet), and two cross-domain test splits: DocTamper-FCD (2k images from the Noisy Office dataset [7]) and DocTamper-SCD (18k images from the HUAWEI Cloud dataset [13]). All images are pre-forged and the dataset supplies pixel-level annotations of tampered regions. The different test sets exhibit substantial domain shift, allowing us to evaluate attack transferability under diverse conditions. To further assess cross-dataset generalization, we additionally evaluate on **Tampered-SROIE** [28], a receipt-based tampered text detection dataset built upon the SROIE corpus [12], containing 986 images (626 training, 360 test) with pixel-level annotations of tampered and authentic text regions.

Models. We evaluate our attacks on six state-of-the-art forgery detectors that rely heavily on DCT-domain cues: CatNet [14], DTD [21], DocForgeNet [23], FFDN [9], RTM [18], and ADCD-Net [29]. These models represent the strongest-performing systems on DocTamper and serve as canonical examples of block-artifact–driven detection pipelines. We apply both proposed attack families, Grid-Aligned Forgeries (GAF-CM, GAF-S, GAF-G) and Pad–Recompress–Crop (PRC), to assess their robustness.

Evaluation Protocol. Following the DocTamper protocol [21], every experimental condition uses 1–3 JPEG recompressions with quality factors ≥ 75 and the public seed. We report

pixel-wise Precision (P), Recall (R), and F1-score (F) on all test splits. To make the comparisons controlled, we keep the underlying forgery type and recompression schedule identical within every baseline–attack pair.

For PRC, which does not create a new forgery, "No Attack/DocTamper Original" refers to the original dataset forgeries and serves as the direct baseline. GAF requires a different comparison because it introduces an additional copy–move, splicing, or generative edit. For each of these forgery types, "No Attack" therefore denotes a paired re-tampering produced without SNAP8. The paired control and GAF branches use the same source and target text instances, forgery operator, mask construction, and recompression schedule; grid alignment is the only intended difference between them.

Attack Evaluation Metrics. For attack-specific evaluation, we define two additional metrics that capture the distinct failure modes induced by GAF and PRC. For image i, let M_i and $\hat{M}_i$ denote the ground-truth and predicted tampered-pixel sets. Detection Failure Rate (DFR) measures how frequently a detector fails to achieve a required amount of localization overlap:

$$\mathrm{IoU}_i = \frac{|M_i \cap \hat{M}_i|}{|M_i \cup \hat{M}_i|}, \qquad \mathrm{DFR}_\tau = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}[\mathrm{IoU}_i \le \tau].$$

We average this quantity over $T = \{0, 0.1, \ldots, 0.5\}$ and report $\mathrm{DFR} = |T|^{-1}\sum_{\tau\in T}\mathrm{DFR}_\tau$; Appendix D presents the individual thresholds. DFR is a sample-level metric, so a higher value means that a larger fraction of images fails to reach the required localization overlap. False Positive Area (FPA) captures the complementary behavior by measuring predictions that fall outside the ground-truth tampered region:

$$\mathrm{FPA} = \frac{1}{N}\sum_{i=1}^{N} \frac{|\hat{M}_i \setminus M_i|}{|\Omega_i|}, \qquad |\Omega_i| = H_i W_i.$$

FPA is normalized by the full image area and then averaged over the dataset. Higher values therefore indicate that spurious tamper predictions cover a larger portion of the evaluated documents.

Implementation Details. Because the datasets contain only pre-forged images, we re-tamper them to construct the paired GAF conditions. We first use EAST [33] to identify text regions, exclude boxes that overlap existing forged areas, and select size-matched source–target pairs. The same selected pair is then used in both branches, and both outputs undergo the common recompression schedule; only the GAF branch applies SNAP8 before the forgery.

For PRC, each image is padded by four pixels on the left and top using edge replication, such that every added pixel takes the value of its nearest boundary pixel, before undergoing the same DocTamper recompression schedule used for the baseline condition. The decoded result is then cropped back to its original dimensions immediately before inference. Consequently, the baseline and PRC conditions share identical compression settings and differ in the half-block grid shift introduced through padding and cropping.

5.2 Quantitative Evaluation: GAF and PRC

In Tables 1 and 2, we present the quantitative evaluation results of our proposed attacks across all detectors and test splits. Here, TestingSet, FCD, and SCD refer to the three **DocTamper** test splits (D-TestingSet, DocTamper-FCD, and DocTamper-SCD). Table 2 additionally reports cross-dataset results on **TSROIE** (Tampered-SROIE), with models trained

Table 1: Performance under PRC and GAF. "DocTamper Original" is the dataset baseline; within each forgery type, "No Attack" is paired re-tampering that differs from GAF only by SNAP8. DFR is shown for GAF settings and FPA for PRC.

			TestingSet				FCD				SCD			
Detection Model	**Attack Type**	**Forgery Type**	**P**	**R**	**F**	**DFR/FPA**	**P**	**R**	**F**	**DFR/FPA**	**P**	**R**	**F**	**DFR/FPA**
CAT-Net	No Attack	DocTamper Original	0.673	0.847	0.750	-	0.774	0.911	0.837	-	0.535	0.835	0.652	-
	PRC	DocTamper Original	0.544	0.731	0.624	0.011	0.568	0.604	0.585	0.042	0.437	0.726	0.546	0.009
	No Attack	Copy-Move	0.766	0.796	0.781	0.230	0.840	0.866	0.853	0.114	0.739	0.836	0.785	0.148
	GAF-CM	Copy-Move	0.744	0.550	0.633	0.570	0.810	0.415	0.549	0.874	0.685	0.524	0.594	0.508
	No Attack	Generative	0.871	0.736	0.798	0.153	0.911	0.839	0.874	0.007	0.869	0.698	0.774	0.083
	GAF-G	Generative	0.871	0.687	0.768	0.202	0.912	0.817	0.862	0.033	0.870	0.650	0.744	0.132
	No Attack	Splicing	0.828	0.846	0.837	0.151	0.839	0.842	0.841	0.114	0.804	0.824	0.814	0.112
	GAF-S	Splicing	0.826	0.669	0.739	0.346	0.828	0.437	0.572	0.774	0.776	0.580	0.664	0.367
DTD	No Attack	DocTamper Original	0.752	0.701	0.726	-	0.793	0.742	0.762	-	0.698	0.701	0.700	-
	PRC	DocTamper Original	0.057	0.823	0.107	0.177	0.084	0.215	0.121	0.084	0.057	0.749	0.105	0.111
	No Attack	Copy-Move	0.877	0.648	0.745	0.125	0.851	0.762	0.804	0.099	0.898	0.744	0.814	0.095
	GAF-CM	Copy-Move	0.446	0.298	0.357	0.512	0.304	0.029	0.054	0.845	0.686	0.381	0.490	0.439
	No Attack	Generative	0.953	0.479	0.638	0.143	0.948	0.782	0.857	0.045	0.942	0.470	0.627	0.172
	GAF-G	Generative	0.514	0.320	0.395	0.417	0.856	0.515	0.643	0.332	0.675	0.312	0.426	0.392
	No Attack	Splicing	0.928	0.752	0.831	0.068	0.839	0.737	0.785	0.101	0.933	0.791	0.856	0.057
	GAF-S	Splicing	0.572	0.472	0.517	0.310	0.237	0.102	0.142	0.754	0.727	0.499	0.592	0.276
DocForgeNet	No Attack	DocTamper Original	0.802	0.751	0.774	-	0.945	0.801	0.822	-	0.701	0.739	0.720	-
	PRC	DocTamper Original	0.050	0.874	0.095	0.215	0.067	0.263	0.106	0.131	0.048	0.810	0.091	0.141
	No Attack	Copy-Move	0.886	0.734	0.803	0.074	0.860	0.843	0.851	0.066	0.900	0.835	0.866	0.044
	GAF-CM	Copy-Move	0.386	0.293	0.333	0.532	0.182	0.017	0.032	0.879	0.578	0.367	0.449	0.452
	No Attack	Generative	0.955	0.550	0.698	0.089	0.946	0.790	0.861	0.028	0.943	0.540	0.687	0.107
	GAF-G	Generative	0.432	0.302	0.355	0.428	0.831	0.457	0.590	0.390	0.563	0.299	0.390	0.401
	No Attack	Splicing	0.930	0.821	0.873	0.045	0.859	0.831	0.845	0.062	0.932	0.854	0.891	0.033
	GAF-S	Splicing	0.495	0.428	0.459	0.353	0.163	0.104	0.127	0.761	0.593	0.469	0.524	0.317
FFDN	No Attack	DocTamper Original	0.873	0.840	0.856	-	0.927	0.905	0.916	-	0.805	0.818	0.812	-
	PRC	DocTamper Original	0.783	0.723	0.752	0.001	0.766	0.661	0.710	0.002	0.747	0.723	0.735	0.001
	No Attack	Copy-Move	0.830	0.801	0.815	0.110	0.888	0.943	0.915	0.024	0.864	0.888	0.876	0.062
	GAF-CM	Copy-Move	0.633	0.485	0.549	0.399	0.369	0.273	0.314	0.645	0.667	0.566	0.613	0.346
	No Attack	Generative	0.890	0.567	0.693	0.183	0.939	0.952	0.946	0.004	0.899	0.623	0.736	0.122
	GAF-G	Generative	0.871	0.483	0.621	0.257	0.932	0.904	0.918	0.027	0.866	0.520	0.650	0.218
	No Attack	Splicing	0.808	0.654	0.723	0.247	0.903	0.942	0.922	0.018	0.821	0.719	0.766	0.193
	GAF-S	Splicing	0.686	0.462	0.552	0.422	0.544	0.421	0.475	0.475	0.675	0.465	0.550	0.421
RTM	No Attack	DocTamper Original	0.745	0.701	0.722	-	0.784	0.699	0.739	-	0.643	0.682	0.662	-
	PRC	DocTamper Original	0.650	0.637	0.644	0.002	0.678	0.560	0.613	0.003	0.605	0.652	0.628	0.003
	No Attack	Copy-Move	0.674	0.641	0.657	0.251	0.702	0.693	0.698	0.228	0.712	0.757	0.734	0.173
	GAF-CM	Copy-Move	0.553	0.453	0.498	0.427	0.460	0.350	0.398	0.531	0.570	0.542	0.556	0.365
	No Attack	Generative	0.876	0.581	0.699	0.157	0.937	0.806	0.867	0.016	0.898	0.662	0.762	0.079
	GAF-G	Generative	0.859	0.522	0.649	0.201	0.928	0.755	0.833	0.028	0.879	0.592	0.708	0.126
	No Attack	Splicing	0.831	0.758	0.793	0.144	0.772	0.791	0.781	0.117	0.845	0.802	0.823	0.108
	GAF-S	Splicing	0.831	0.728	0.776	0.159	0.676	0.584	0.627	0.269	0.843	0.770	0.805	0.126
ADCD-Net	No Attack	DocTamper Original	0.789	0.823	0.806	-	0.866	0.770	0.815	-	0.690	0.799	0.740	-
	PRC	DocTamper Original	0.827	0.864	0.845	0.003	0.853	0.634	0.728	0.004	0.631	0.730	0.677	0.004
	No Attack	Copy-Move	0.815	0.498	0.618	0.154	0.930	0.638	0.757	0.115	0.828	0.604	0.698	0.221
	GAF-CM	Copy-Move	0.782	0.403	0.532	0.219	0.897	0.288	0.436	0.291	0.813	0.484	0.607	0.306
	No Attack	Generative	0.926	0.544	0.685	0.137	0.986	0.953	0.970	0.041	0.914	0.608	0.730	0.128
	GAF-G	Generative	0.903	0.424	0.577	0.211	0.989	0.896	0.940	0.005	0.889	0.426	0.575	0.249
	No Attack	Splicing	0.892	0.489	0.632	0.183	0.938	0.701	0.802	0.075	0.900	0.582	0.707	0.178
	GAF-S	Splicing	0.877	0.397	0.547	0.253	0.917	0.466	0.618	0.167	0.894	0.497	0.639	0.240

Table 2: Cross-dataset evaluation on Tampered-SROIE with models trained on DocTamper. "No Attack" is paired re-tampering without SNAP8.

		No Attack				GAF Attack			
Model	**Forgery**	**P**	**R**	**F**	**DFR**	**P**	**R**	**F**	**DFR**
CAT-Net	Copy-Move	0.202	0.168	0.184	0.813	0.106	0.062	0.078	0.951
	Generative	0.302	0.208	0.246	0.702	0.281	0.154	0.199	0.770
	Splicing	0.177	0.103	0.130	0.883	0.122	0.056	0.077	0.952
DTD	Copy-Move	0.867	0.633	0.732	0.742	0.597	0.039	0.073	0.936
	Generative	0.927	0.327	0.483	0.766	0.871	0.144	0.247	0.859
	Splicing	0.875	0.366	0.516	0.794	0.744	0.096	0.171	0.908
DocForgeNet	Copy-Move	0.921	0.844	0.881	0.690	0.830	0.032	0.062	0.960
	Generative	0.972	0.619	0.757	0.687	0.986	0.209	0.345	0.837
	Splicing	0.925	0.631	0.751	0.722	0.857	0.032	0.062	0.961
FFDN	Copy-Move	0.201	0.183	0.192	0.792	0.090	0.062	0.074	0.920
	Generative	0.296	0.159	0.207	0.746	0.284	0.127	0.175	0.782
	Splicing	0.153	0.096	0.118	0.869	0.080	0.035	0.049	0.946
RTM	Copy-Move	0.219	0.227	0.223	0.747	0.077	0.056	0.065	0.925
	Generative	0.305	0.207	0.246	0.696	0.297	0.170	0.216	0.726
	Splicing	0.176	0.129	0.149	0.828	0.110	0.057	0.075	0.914
ADCD-Net	Copy-Move	0.735	0.747	0.741	0.683	0.335	0.235	0.276	0.830
	Generative	0.854	0.493	0.625	0.688	0.803	0.302	0.439	0.760
	Splicing	0.538	0.381	0.447	0.759	0.270	0.159	0.200	0.849

on DocTamper. Following the evaluation protocol described in Sect. 5, we compare each model under the standard "No Attack" tampering setup against the three variants of Grid-Aligned Forgery (GAF-CM, GAF-S, GAF-G) and the Pad–Recompress–Crop (PRC) attacks. We summarize our key observations below.

GAF-CM and GAF-S. Across the evaluated datasets and splits, GAF-CM and GAF-S produce substantial yet model-dependent degradation. On TestingSet Copy-Move, for example, CAT-Net falls from 0.781 to 0.633 F1, a relative decline of approximately 19%, while RTM falls from 0.657 to 0.498, a decline of approximately 24%. The effect is even stronger for several models on FCD: CAT-Net, RTM, and ADCD-Net decrease from 0.853 to 0.549, 0.698 to 0.398, and 0.757 to 0.436, respectively. GAF-S produces a similar pattern, including CAT-Net's FCD decline from 0.841 to 0.572 and DTD's decline from 0.785 to 0.142.

The DFR results show that these F1 reductions correspond to failures on a substantial fraction of samples rather than only minor boundary errors. On TestingSet, CAT-Net's DFR rises from 0.230 to 0.570 under GAF-CM and from 0.151 to 0.346 under GAF-S. Under GAF-CM, DTD rises from 0.125 to 0.512 on TestingSet and from 0.099 to 0.845 on FCD; DocForgeNet similarly rises from 0.074 to 0.532 and from 0.066 to 0.879. The same vulnerability transfers across datasets. On TSROIE Copy-Move, DTD changes from $F/\mathrm{DFR} = 0.732/0.742$ to $0.073/0.936$, while DocForgeNet changes from $0.881/0.690$ to $0.062/0.960$ (Table 2). Together, these results support our central hypothesis that preserving the JPEG grid can remove cues on which many current detectors strongly depend.

GAF-G. Compared with copy–move and splicing, GAF-G generally produces milder degradation, although the response varies considerably across architectures. On TestingSet, CAT-Net changes from 0.798 to 0.768 F1, FFDN from 0.693 to 0.621, RTM from 0.699 to 0.649, and ADCD-Net from 0.685 to 0.577. DTD and DocForgeNet are more strongly affected, falling from 0.638 to 0.395 and from 0.698 to 0.355, while their DFR values rise from 0.143 to 0.417 and from 0.089 to 0.428, respectively.

This weaker and less uniform effect is consistent with the nature of generative tampering. Rendered text can retain mismatched quantization signatures, antialiasing patterns, and font-

texture statistics even after grid alignment, leaving additional cues that may help a detector localize the edit. Grid alignment alone therefore does not remove every artifact introduced by generative content.

PRC. PRC produces a characteristic failure pattern in the FPA column. DTD and DocForgeNet show the clearest false-positive inflation, reaching FPA values of 0.177 and 0.215 on TestingSet while their F1 scores fall from 0.726 to 0.107 and from 0.774 to 0.095. This behavior also transfers to FCD, where their FPA values reach 0.084 and 0.131. CAT-Net, FFDN, and RTM avoid such extensive over-detection, but their TestingSet F1 scores still decline from 0.750 to 0.624, 0.856 to 0.752, and 0.722 to 0.644, respectively. The response is not universal: ADCD-Net records only 0.003 FPA and changes from 0.806 to 0.845 F1 on TestingSet.

To better understand FFDN's relative stability, Appendix C reports an ablation in which its WFE module is removed. The small difference between the two configurations indicates that WFE alone is unlikely to explain the observed behavior. Other components, including VEM-mediated fusion, may contribute, but this experiment does not isolate their effects and should not be interpreted as a complete architectural or defense study.

Taken together, GAF and PRC should be viewed as complementary diagnostic probes rather than conventional optimization-based attacks. Without querying the detector, they expose two opposing failure modes: forged regions may be missed when the JPEG grid is preserved, whereas false alarms or degraded localization may arise when it is shifted. The consistency of these patterns across architectures and datasets provides strong evidence of JPEG-grid dependence and motivates document-forensics representations that are both semantically grounded and robust to compression structure. Appendix B places this threat model in context by comparing it with prior white-box attacks.

5.3 Qualitative Analysis: GAF and PRC

The qualitative results in Figs. 4 and 5 illustrate the failure modes summarized by the quantitative metrics. Under GAF-CM and GAF-S, DTD and DocForgeNet often produce nearly empty masks relative to the paired unaligned controls, even though the inserted text remains visually apparent. CAT-Net and RTM preserve more of the manipulated region but still yield incomplete masks, whereas FFDN is comparatively stable in several examples. These observations are consistent with the strong DFR increases reported in Tables 1 and 2.

PRC exposes the complementary behavior. DTD and DocForgeNet incorrectly label extensive non-forged regions as tampered, in line with their high FPA values in Table 1. CAT-Net, FFDN, RTM, and ADCD-Net generally avoid such widespread false positives, although their localization masks can still change or become less complete. Taken together, the examples show how preserving and shifting the JPEG grid can lead to missed detections and false alarms, respectively.

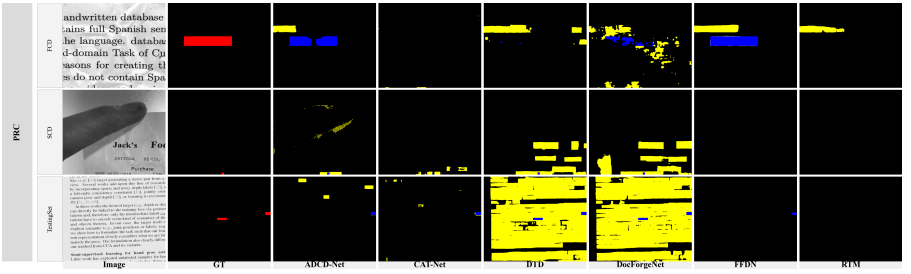


Figure 4: Qualitative comparison of PRC across document-tampering detectors on DocTamper [21]. DTD and DocForgeNet exhibit widespread false positives, whereas the remaining models show smaller but visible changes in their predictions. **Red** denotes the ground truth, **blue** denotes true positives, and **yellow** denotes false positives.

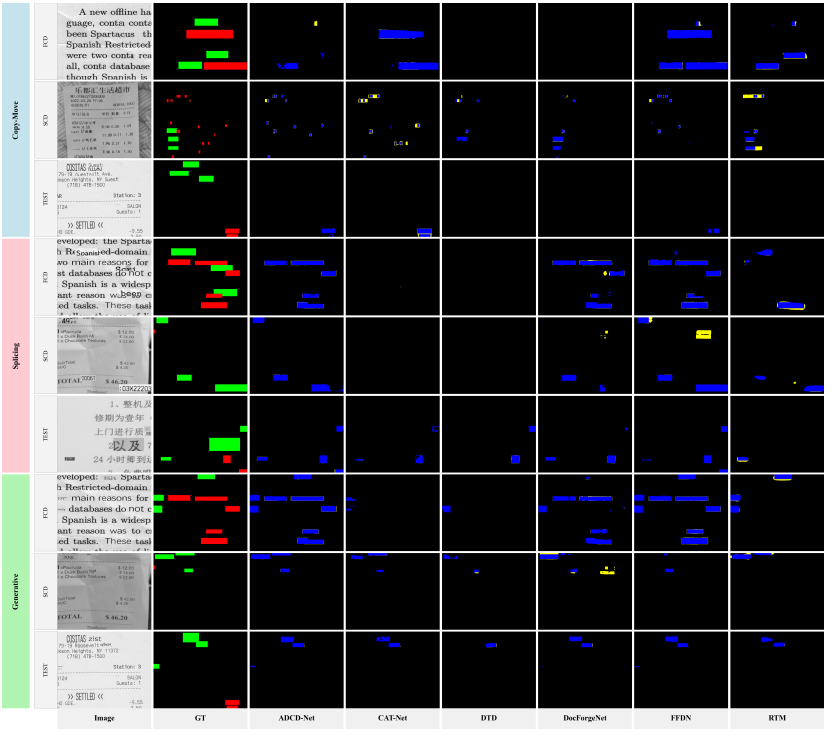


Figure 5: Qualitative comparison of GAF across document-tampering detectors on DocTamper [21]. Relative to paired unaligned re-tamperings, GAF often yields nearly empty predictions, particularly for copy–move and splicing. **Green** denotes the GAF mask, **red** the original ground truth, **blue** true positives, and **yellow** false positives.

6 Limitations

While our attacks expose significant vulnerabilities in current forgery detectors, several limitations warrant discussion. First, the study is deliberately restricted to JPEG-based

RGB+DCT models. Lossless and non-JPEG pipelines do not contain the block-quantization structure targeted by GAF and PRC, and evaluating frequency-independent document detectors remains an important direction for future work. GAF also assumes knowledge of the JPEG grid origin, as do common uncropped copy–move benchmarks. If earlier cropping obscures that origin, an attacker would first need to estimate it; PRC does not require this information because its fixed shift is applied globally.

Second, neither intervention perfectly isolates JPEG-grid statistics. SNAP8 can expand each box boundary by fewer than eight pixels, leaving a small geometric difference between GAF and its paired control. PRC preserves semantic content, image dimensions, and spatial layout, but recompression on shifted blocks may slightly alter decoded RGB values. We evaluate PRC using the fixed $(4,4)$ shift; evaluating the complete shift space remains future work. The controlled comparisons make grid sensitivity the most direct explanation for the observed failures, but they cannot exclude every residual RGB or geometric effect.

7 Conclusions

We introduced GAF and PRC as complementary, query-free probes of JPEG-grid bias in document forgery detectors. GAF aligns copy–move, splicing, and generative edits with the JPEG grid, while PRC applies a fixed half-block grid shift without changing semantic content or image geometry. Across six detectors and two document benchmarks, the former frequently causes missed forged regions and the latter causes false alarms or degraded localization, although vulnerability varies by architecture and forgery type.

These findings show that strong benchmark performance can coexist with sensitivity to compression structure that is incidental to the document's semantic integrity. They also expose practical reliability risks: an aligned manipulation may evade localization, whereas benign variation in JPEG-grid position may reduce confidence or generate false detections. GAF and PRC provide reproducible stress tests for this behavior. More reliable document forensics will require representations and training procedures that remain stable across compression histories while retaining semantically grounded evidence of manipulation.

Acknowledgements

This work is partially supported by the InnoTop, Programme of the State of Rhineland-Palatinate, funded by the European Regional Development Fund.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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*, pages 1687–1690, 2010. doi: 10.1109/ISCAS.2010.5537505.

[5] 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.

[6] 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.

[7] 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.

[8] Yi-Lei Chen and Chiou-Ting Hsu. Image tampering detection by blocking periodicity analysis in jpeg compressed images. In *2008 IEEE International Workshop on Multimedia Signal Processing*, pages 803–808, 10 2008. doi: 10.1109/MMSP.2008.4665184.

[9] 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, editors, *Computer Vision – ECCV 2024*, pages 200–217, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73414-4.

[10] 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.

[11] 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.

[12] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. ICDAR 2019 competition on scanned receipts OCR and information extraction. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520, 2019. doi: 10.1109/ICDAR.2019.00244.

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

[14] 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*, pages 375–384, 2021.

[15] 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.

[16] 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.

[17] 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.

[18] Dongliang Luo, Yuliang Liu, Rui Yang, Xianjin Liu, Jishen Zeng, Yu Zhou, and Xiang Bai. Toward real text manipulation detection: New dataset and new solution. *Pattern Recognition*, 157:110828, 2025. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2024.110828. URL https://www.sciencedirect.com/science/article/pii/S003132032400579X.

[19] 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.

[20] 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.

[21] 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)*, pages 5937–5946, June 2023.

[22] 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.

[23] 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, editors, *Document Analysis and Recognition – ICDAR 2025*, pages 329–346, Cham, 2025. Springer Nature Switzerland.

[24] Huiru Shao, Zhuang Qian, Kaizhu Huang, Wei Wang, Xiaowei Huang, and Qiufeng Wang. Delving into adversarial robustness on document tampering localization. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 290–306, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73650-6.

[25] 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.

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

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

[28] 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, 2022. doi: 10.11959/j.issn.2096-109x.2022035. URL https://www.infocomm-journal.com/cjnis/EN/abstract/article_172502.shtml.

[29] Kahim Wong, Jicheng Zhou, Haiwei Wu, Yain-Whar Si, and Jiantao Zhou. Adcd-net: Robust document image forgery localization via adaptive dct feature and hierarchical content disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19280–19289, October 2025.

[30] 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*, pages 1500–1508, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3343031.3350929. URL https://doi.org/10.1145/3343031.3350929.

[31] 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.

[32] 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.

[33] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: An efficient and accurate scene text detector, 2017. URL https://arxiv.org/abs/1704.03155.