READING THE VISUAL CLUES: A DATASET AND CONTEXTUAL DETECTOR FOR DOCUMENT TAMPERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Document tampering extends beyond edited text to figures, formulas, and tables, motivating a common evaluation of editing operations across page elements. We introduce MEDTD, a benchmark for localizing operation-involved regions and recognizing editing operations from a single full-page image. Constructed from DocLayNet, MEDTD contains 46,290 programmatically edited pages, 25,145 untampered counterparts, and 69,839 region annotations covering seven element groups and four operations. Its annotations include unchanged duplication sources and vacated relocation sources, rather than modified pixels alone. We further introduce ClueDet, a DINO-based detector that refines detection queries through similarity-weighted comparisons of predicted regions. Under matched training settings, ClueDet achieves 34.11 operation AP, compared with 33.02 for DINO and 32.56 for a parameter-matched uniform-context variant. Improvements are concentrated in textual elements and splicing rather than shared uniformly across element groups. Evaluation on untampered pages and external datasets characterizes false-positive behavior and limits to transfer. MEDTD provides a benchmark for studying operation-aware document tampering across heterogeneous page elements.

1 INTRODUCTION

Document tampering can affect both the content of a region and its relationship to other elements on the page. A duplication event involves an unchanged source and an inserted copy, whereas relocation involves a vacated source and a destination. Identifying only modified pixels therefore does not fully describe the regions involved in these operations. We study operation-aware region detection from a document page image. The detector must localize the regions associated with an editing operation and assign an operation label without access to the original page or editing metadata. Applying this task to words, lines, paragraphs, figures, formulas, and tabular content enables comparison across editing units with different spatial extents and visual structures.

Recent document forgery datasets provide large collections of edited text and increasingly detailed annotations of manipulation techniques (Qu et al., 2023; Yu et al., 2025; Qin et al., 2026; Qu et al., 2026b). A common evaluation of editing operations across textual and visual elements remains less developed. Words, formulas, figures, and tables differ in size and internal structure. Their surroundings provide evidence of copying, removal, and replacement. A dataset covering these elements under a shared annotation scheme is needed to study detection beyond text.

We introduce MEDTD, the Multi-Element Document Tampering Dataset. MEDTD comprises 46,290 tampered pages, 25,145 untampered pages, and 69,839 region annotations. It covers seven element groups and four editing operations. The task is to localize the regions involved in an edit and identify its operation from a single page image. Annotations include both regions of duplication and relocation events, including an unchanged duplication source or a vacated relocation source.

MEDTD provides complete pages and their untampered counterparts, preserving relationships between edited content and surrounding elements. Untampered pages enable evaluation of false positives caused by ordinary repetition and layout variation. Partitions group all pages and edited deriva-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

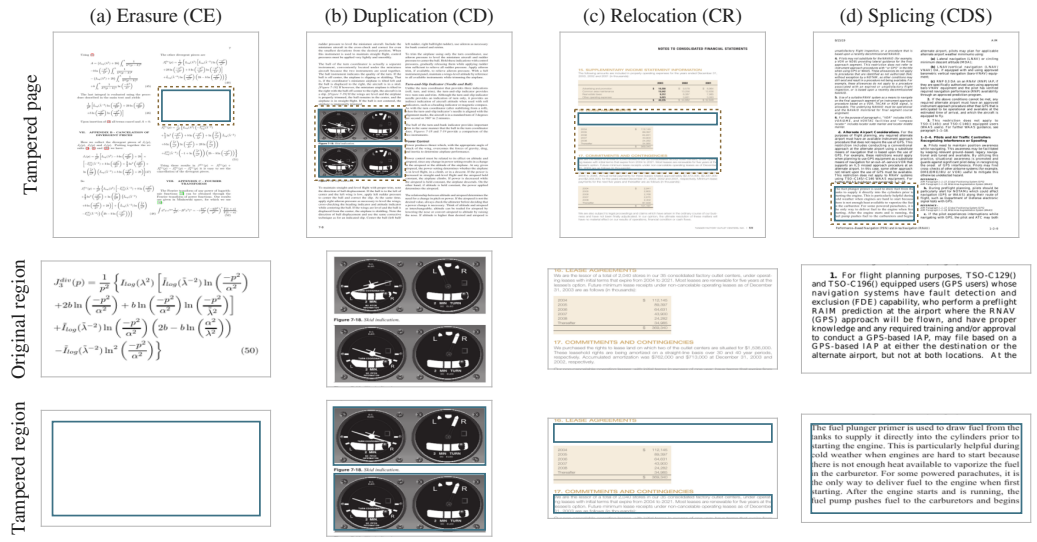


Figure 1: **Document tampering across text and visual elements.** Each column shows a MEDTD training page and enlarged regions before and after an edit. Solid boxes mark detection targets. Dashed boxes delimit the crops. Duplication includes the source and copy, while relocation includes the vacated source and destination. The detector receives only the tampered page. Original regions are shown to illustrate the editing operation.

tives of a recipient document, so the test set contains different recipient documents from those used for training.

We further propose ClueDet, a contextual detector built on DINO (Zhang et al., 2023). ClueDet samples features within predicted boxes and aggregates features from other regions according to visual similarity. A residual update then combines the current region with its difference from the aggregated context. The module shares parameters across the final two decoder insertions and predicts the same four operation classes as the baseline. Its region comparisons require neither original pages nor editing-role labels at inference.

The experiments examine training coverage and contextual aggregation separately. At a fixed optimization budget, training on the full set of pages improves operation AP by 12.16 points for DINO and 1.86 points for Faster R-CNN over fixed half-data subsets. ClueDet achieves 34.11 operation AP, compared with 33.02 for DINO and 32.56 for a parameter-matched uniform-context model. These results support broader training coverage and similarity-weighted regional comparison, with the latter's gains concentrated in text and splicing. The contributions of our work are as follows:

- We introduce MEDTD for detecting four editing operations across textual and visual document elements. The dataset provides full pages, operation-labeled regions, untampered counterparts, and partitions grouped by recipient document.
- We propose ClueDet, which refines detection queries through similarity-weighted regional aggregation and the difference between contexts.
- We establish the proper comparisons, which quantify the effects of distinct-page coverage and similarity-weighted aggregation under the studied settings.

2 Related Work

Document tampering datasets. STFD studies forged screenshot text, while DocTamper provides a large collection of manipulated document text (Yu et al., 2023; Qu et al., 2023). RTM includes tampered text from documents and scenes (Luo et al., 2025). Subsequent datasets expand the editing processes and annotations. FSTS models traces from human editing, FGDTD distinguishes manip-

Dataset	Year	Document Type	Docs.	Original images	Tampered images	Edited Elements
STFD	2023	Screenshots	NR	NR	4,094	Text
DocTamper	2023	Business records	NR	50,562	170,000	Text
RTM	2025	Documents / scenes	NR	3,000	6,000	Text
FSTS	2025	Scans / captures	NR	NR	294,182	Text
FGDTD	2026	Document text	NR	NR	16,479	Text
DanceText	2026	Documents / scenes	NR	144,657	793,731	Text
AIForge-Doc†	2026	Receipts / forms	NR	4,061	4,061	Numeric text
MEDTD	2026	Mixed documents	1,995	25,145	46,290	Text, graphics, tables

Table 1: **Document and text-image tampering datasets.** Years follow the cited publications, with † marking a preprint. Docs. counts source-document identities, while image counts follow each dataset's reporting unit. NR denotes an unreported count. DanceText includes fully generated images. AIForge-Doc edits individual numeric fields. MEDTD groups edited content into seven elements spanning text, graphics, and tables.

ulation techniques, and DanceText includes AI-generated and edited text (Yu et al., 2025; Qin et al., 2026; Qu et al., 2026b). DeepID and AIForge-Doc focus on identity documents and generative field edits (Korshunov et al., 2025; Wu et al., 2026). Table 1 summarizes their scope and scale. MEDTD complements these resources by applying a common four-operation task to text, figures, formulas, and tabular elements within full pages.

Visual document forensics. Document forensic methods use appearance, frequency, and content features to distinguish edited regions. ADCD-Net adapts frequency features to document content, DCL-Net addresses document desensitization, and TextShield-R1 uses reasoning for tampered-text detection (Wong et al., 2025; Li et al., 2026; Qu et al., 2026a). TruFor jointly considers localization and prediction reliability (Guillaro et al., 2023). IMDL-BenCo and ForensicHub provide shared implementations for image manipulation detection (Ma et al., 2024; Du et al., 2025). These methods differ in their output representation and training targets. MEDTD evaluates region localization together with operation recognition and false positives on untampered pages. The latter is relevant to recent work on decision thresholds and document forgery evaluation (Yang et al., 2026; Zhao et al., 2026).

Region and document context. ObjectFormer models consistency between object and patch representations, and Forensim compares features to localize source and target regions (Wang et al., 2022; Nandi & Natarajan, 2026). GraphDoc studies relations between document elements, while RoDLA and OmniDocBench address layout analysis and document parsing (Chen et al., 2025; 2024; Ouyang et al., 2025). ClueDet uses predicted regions as the units of comparison within a detection decoder. The uniform-context control retains the same architecture and parameter count to examine the effect of similarity weighting. Separately, our training-coverage study compares different page sets at matched update counts. It complements studies of generator diversity at fixed image counts, such as Community Forensics (Park & Owens, 2025).

3 METHODS

3.1 THE MEDTD DATASET

MEDTD pairs four editing operations with seven groups of document elements. Operations define the detection classes, while elements identify the edited content. This organization supports comparisons across text, graphics, and tables within complete pages.

3.1.1 TASK AND REGION ANNOTATIONS

Given an RGB page I, a detector predicts $\widehat{Y}(I) = \{(\hat{b}_j, \hat{y}_j, \hat{s}_j)\}_{j=1}^{m}$, where $\hat{b}_j$ is a bounding box, $\hat{y}_j$ is an operation label, and $\hat{s}_j$ is its confidence. The reference is $Y = \{(b_i, y_i)\}_{i=1}^{n}$. Untampered

Element	Editing unit	CE	CD	CR	CDS	Pages	Regions
Word	Detected word	2,499	2,341	2,499	1,721	9,060	13,900
Line	Grouped words	2,047	2,103	2,111	1,814	8,075	12,289
Paragraph	Text block	2,256	2,224	2,099	1,912	8,491	12,814
Picture	Picture region	2,203	920	1,282	534	4,939	7,141
Formula	Formula region	2,394	833	1,266	669	5,162	7,261
Table	Whole table	1,954	977	766	265	3,962	5,705
Table-cell content	Text within a table	2,171	2,104	2,024	302	6,601	10,729
Total		15,524	11,502	12,047	7,217	46,290	69,839

Table 2: **Element counts for the four tampering operations in MEDTD.** Operation columns count editing events across all partitions. Each event produces one tampered page. Regions counts all target boxes, including both regions of duplication and relocation. Table-cell content denotes local text candidates within annotated tables.

pages have $Y = \varnothing$. Inference uses only the page image. Original pages and editing metadata support dataset construction and analysis.

Each tampered page contains one editing event. Content Erasure (CE) removes a selected region. Content Duplication (CD) copies it within the same page. Content Relocation (CR) moves it to another location. Cross Document Splicing (CDS) inserts content from a different page. Erasure and splicing have one target region. Duplication annotates both the unchanged source and its copy, while relocation annotates the vacated source and destination. Both targets share the operation label. The targets are operation-involved regions rather than a mask of modified pixels. In particular, duplication includes an unchanged source region, and relocation includes a vacated source region. Both boxes are evaluated independently under the operation label, while not requiring predicting a source–destination pairing or copying direction.

3.1.2 Element taxonomy

The seven groups organize candidates by spatial extent and document role. Words, lines, and paragraphs cover three granularities of text editing. Pictures and formulas cover graphical and mathematical content with distinct internal structures. Tables and table-cell content distinguish whole-table edits from local changes within tabular regions. An element label describes the candidate selected for an event. It guides construction and evaluation.

Words, lines, and paragraphs form the Text subset. Pictures, formulas, tables, and table-cell content form the Visual subset. This assignment follows the edited element rather than the contents of the surrounding document. All seven groups contain examples of all four operations. Nine retained annotation tags map to these groups, including alternate names for lines and whole tables. This gives 28 semantic operation–element combinations and 30 combinations under the original tags.

3.1.3 Dataset creation

We construct edited derivatives from DocLayNet page images and layout annotations (Pfitzmann et al., 2022). A pretrained text detector supplies word candidates. Grouping detections with compatible vertical positions and heights produces line candidates. Paragraph candidates use DocLayNet text-block boxes, while its picture, formula, and table boxes supply the corresponding visual candidates. Text detection within annotated table regions provides the local candidates reported as table-cell content.

The editing routines act on these candidate regions. Erasure fills the selected rectangle with a background color estimated from the page. Duplication pairs source and receiving regions with comparable dimensions and copies the source content. Relocation additionally fills the source region with the estimated background. Splicing inserts content extracted from a second page into the recipient region. Figure 1 illustrates the resulting edits. Each derivative retains the complete page and is stored as a 1025×1025 PNG with operation labels, element tags, region boxes, and its original-page association.

Partition	Tampered pages	Region boxes	Untampered pages	Documents
Train	32,403	48,887	17,892	1,397
Validation	4,629	6,984	2,404	199
Test	9,258	13,968	4,849	399
Total	46,290	69,839	25,145	1,995

Table 3: **MEDTD partitions.** All original pages and tampered derivatives of a recipient document remain in one split. Original pages are counted once per distinct source image.

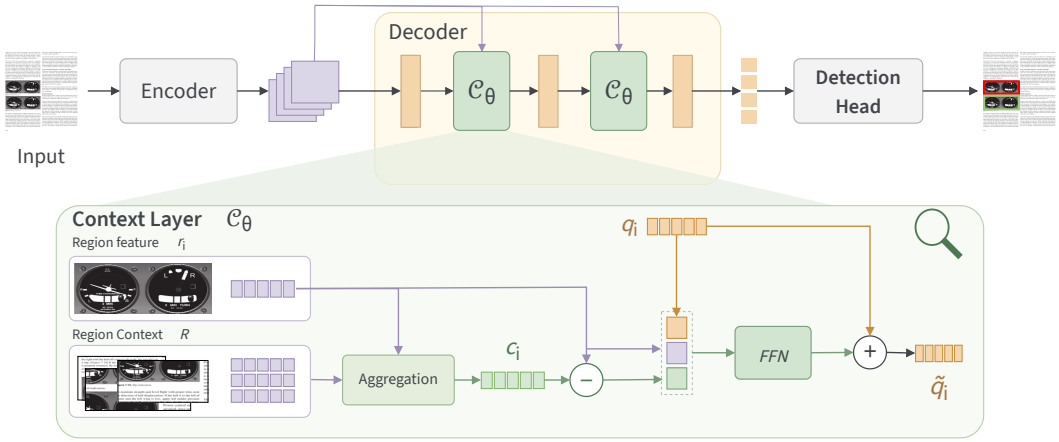


Figure 2: **ClueDet Structure Illustration.** Context layers before decoder layers 5 and 6 share parameters and encoder memory. The enlarged view shows how the current region feature r_i and region bank R form context c_i. A feed-forward network (FFN) projects $[q_i; r_i; c_i - r_i]$ to a residual query update. Each insertion recomputes region features from the current predicted boxes. Crops, tokens, and output boxes illustrate the information flow.

After the dataset creation, every generated page, corrected editing and annotation defects, was checked and modified by human experts. Across the 46,290 tampered pages, the retained boxes contain 99.715% of reconstructed changed pixels.

3.1.4 Composition and Partitions

MEDTD contains 46,290 tampered pages, 25,145 distinct original pages, and 69,839 target boxes from 1,995 recipient documents. Table 2 gives the number of edited pages and annotated regions for every element group. Derivatives can share an original page. The source domains comprise financial reports, scientific articles, laws and regulations, manuals, patents, and government tenders.

Table 3 partitions pages by collection and recorded recipient-document identity. All pages and edited derivatives of a recipient document remain together. Validation selects model configurations and decision thresholds. The common test set contains 9,258 tampered pages for localization and operation recognition, together with 4,849 untampered pages for false-positive evaluation. The images, annotations, metadata, partitions, and data card follow CDLA-Permissive-1.0 with upstream attribution and modification notices.

3.2 ClueDet

Repeated or displaced content provides evidence that is not confined to a single region. ClueDet extends DINO with a regional-context module that compares features within predicted boxes and refines the corresponding detection queries. Figure 2 shows the two insertions before decoder layers 5 and 6. Region similarity supplies contextual evidence that the detection head combines with the query and regional appearance to determine the operation and location.

Region features. For a decoder query $q_i \in \mathbb{R}^{256}$ with normalized box b_i, we sample the centers of a 3×3 grid within b_i at each of four encoder-memory scales. We average the samples across positions and scales and apply parameter-free LayerNorm to obtain the region feature r_i. Box coordinates are detached during sampling, while gradients are retained through the image features.

The region bank $R = \{r_j\}_{j=1}^{N}$ contains features at the current predicted boxes of all matching queries. Denoising queries are excluded from the region bank. The feature r_i of the current matching query is also its entry in this bank. The admissible set $\mathcal{A}_i$ excludes that entry and regions whose boxes overlap b_i with IoU at least 0.5. This restriction reduces repeated sampling of the same location.

Context aggregation. Cosine similarity determines the contribution of each admissible region,

$$a_{ij} = \frac{\exp(\cos(r_i, r_j)/\tau)}{\sum_{k\in\mathcal{A}_i} \exp(\cos(r_i, r_k)/\tau)}, \qquad c_i = \sum_{j\in\mathcal{A}_i} a_{ij} r_j, \tag{1}$$

Here $j \in \mathcal{A}_i$ and $\tau = 0.2$. The weighted context c_i constructs an appearance reference from admissible regions relative to r_i on the same page. Their difference $c_i - r_i$ represents the comparison used to update the query. When $\mathcal{A}_i$ is empty, we set $c_i = r_i$. The detection head learns how regional appearance and its relation to this reference contribute to localization and operation recognition.

Query refinement. The query is updated through a residual projection of its representation, regional appearance, and contextual difference,

$$\widetilde{q}_i = q_i + f_\theta\big([q_i, r_i, c_i - r_i]\big). \tag{2}$$

The projection f_θ is a $768 \to 256 \to 256$ linear–GELU–linear network with 262,656 parameters. Its output layer is initialized to zero, preserving the initial queries. Both insertions share f_θ and the encoder memory. Region features and aggregation weights are recomputed from the boxes available at each insertion.

ClueDet retains DINO's classification and box losses and predicts all four operations with one detection head. It uses all 32,403 tampered training pages with DINO's initialization, image order, and schedule. Supervision consists of the four operation labels and target boxes. The uniform-context control replaces a_{ij} with $1/|\mathcal{A}_i|$ and retains the sampling, masking, projections, and insertion layers. This comparison tests similarity weighting at the same parameter count.

3.3 Evaluation metrics

Operation AP evaluates localization and operation recognition jointly. We use COCO-style average precision over the four classes and IoU thresholds from 0.50 to 0.95 in increments of 0.05. Region AP pools the operation labels to evaluate localization alone. Text AP covers words, lines, and paragraphs on 5,319 tampered test pages. Visual AP covers pictures, formulas, tables, and table cells on the remaining 3,939. These metrics use the same predictions and retain their coordinates and confidence scores.

Untampered-page FPR is the fraction of pages with a maximum prediction score of at least 0.5. It evaluates false positives separately from AP on tampered pages. Box detectors retain scores of at least 0.001 and at most 100 predictions per page. For binary models, probability maps are restored to page resolution and thresholded at 0.5. Each eight-connected component yields a box scored by its mean probability, and the 100 highest-scoring boxes enter region evaluation. The page score is the maximum pixel probability. Binary outputs are evaluated through region localization and false positives.

4 Experiments

4.1 Compared methods

We compare object detectors, fusion methods, forensic localizers, and a vision-language model on the same 9,258 tampered and 4,849 untampered test pages. Detection baselines include Faster

Family	Method	Operation AP					Class-agnostic AP			FPR↓
		Mean↑	CE↑	CD↑	CR↑	CDS↑	All↑	Text↑	Visual↑	
Detection	Faster R-CNN	10.55	23.66	2.47	12.75	3.30	17.24	12.36	24.61	5.65
	RetinaNet	12.46	25.61	2.71	14.17	7.36	18.07	14.28	23.43	**0.49**
	Cascade R-CNN	13.70	25.22	2.83	14.33	12.42	19.61	15.07	26.28	11.61
	RTMDet-M	17.07	29.71	4.32	16.28	17.98	20.21	16.76	25.39	18.21
	DINO	33.02	40.93	12.74	35.19	43.21	34.17	30.76	**39.22**	8.79
	ClueDet (Uniform) (Ours)	32.56	39.89	11.97	33.94	44.45	34.00	30.97	38.46	9.84
	ClueDet (Ours)	**34.11**	41.02	**13.24**	**35.83**	**46.35**	**35.13**	**33.01**	38.83	9.53
Fusion	WBF	29.38	38.07	9.17	27.98	42.32	30.77	27.45	35.67	12.72
	Proposal alignment	33.13	**41.25**	12.45	35.62	43.19	34.06	30.72	39.14	9.57
Forensics	DCL-Net	—	—	—	—	—	13.62	9.17	19.96	22.21
	ADCD-Net	—	—	—	—	—	0.99	0.10	0.99	51.12
	Co-Transformers	—	—	—	—	—	0.02	<0.01	0.07	69.48
	CAFTB-Net	—	—	—	—	—	9.90	6.29	15.20	20.71
	TIFDM	—	—	—	—	—	9.13	5.52	14.19	12.11
	FRD-Net	—	—	—	—	—	13.16	9.97	17.87	90.55
	CAT-Net v2	—	—	—	—	—	3.85	3.75	4.21	12.44
	PSCC-Net	—	—	—	—	—	3.20	1.88	5.22	86.53
	TruFor	—	—	—	—	—	0.09	<0.01	0.31	100.00
	FLAME	—	—	—	—	—	6.40	3.82	10.11	33.06
Vision-language	PIXAR	—	—	—	—	—	0.61	0.01	0.64	100.00

Table 4: **Document tampering detection on MEDTD.** AP is evaluated on 9,258 tampered test pages and reported on a 0–100 scale. FPR (%) is evaluated separately on 4,849 untampered test pages at score threshold 0.5. The class-agnostic All column reports region AP. Binary outputs do not predict operation classes (—). Bold marks the best result in each column. Training settings and output conversions follow Sections 4.2 and 3.3.

R-CNN, RetinaNet, Cascade R-CNN, RTMDet, Deformable DETR, and DINO (Ren et al., 2015; Lin et al., 2017; Cai & Vasconcelos, 2018; Lyu et al., 2022; Zhu et al., 2021; Zhang et al., 2023). Weighted boxes fusion combines detector predictions (Solovyev et al., 2021). The Proposal alignment baseline combines detector boxes with DCL-Net proposals through semantic–forensic alignment.

Adapted forensic baselines include PSCC-Net, FRD-Net, FLAME, CAT-Net, TIFDM, and CAFTB-Net (Liu et al., 2022; Chen et al., 2026; Wang et al., 2026; Kwon et al., 2022; Dong et al., 2024; Song et al., 2024). ADCD-Net, Co-Transformers, TruFor, and PIXAR use released weights without MEDTD adaptation (Zhang et al., 2026; Shang et al., 2026). MEDTD adaptation uses all 32,403 tampered training pages. The newly adapted forensic baselines additionally use 4,445 pages selected from the 17,892-page untampered training partition, with box-union supervision. Each method retains its initialization, input resolution, and supervision. Architectural attribution is based on the matched DINO context comparisons.

4.2 Implementation details

DINO, ClueDet, and the uniform-context control use an ImageNet-initialized ResNet-50 backbone and a four-scale encoder. Training lasts 12 epochs over all tampered training pages. Whole pages are resized to fit 1333×800 without crops or flips. AdamW uses a learning rate of 10^{-4}, a backbone multiplier of 0.1, and weight decay of 10^{-4}. Accumulation over two pages gives 194,418 optimizer updates. All three models use their final seed-0 checkpoints. The half-data study uses 16,202 pages with the same initialization, update count, and learning-rate schedule. Faster R-CNN provides a second architecture for examining training coverage. Validation comprises 4,629 tampered and 2,404 untampered pages. For training-coverage and context comparisons, 95% intervals use 2,000 paired bootstrap samples over the 399 test documents.

Method	Training set	Aggregation	Op. AP↑	Region AP↑	FPR↓
Faster R-CNN	Half	—	8.68	14.55	16.81
	Full	—	**10.55**	**17.24**	**5.65**
DINO	Half	—	20.86	25.58	28.81
	Full	—	33.02	34.17	**8.79**
ClueDet	Full	Uniform	32.56	34.00	9.84
	Full	Similarity	**34.11**	**35.13**	9.53

Table 5: **Training-coverage and regional-aggregation ablations.** Half and Full contain 16,202 and 32,403 training pages, respectively, covering the same documents and operation–tag combinations. Initialization and optimization budgets are matched within each comparison. The two ClueDet variants differ only in the aggregation rule; "—" denotes no added regional-context module. AP uses a 0–100 scale; FPR (%) uses score threshold 0.5. Bold marks the best result in each block.

4.3 RESULTS

Document tampering detection. Table 4 compares operation recognition and region localization across detection, fusion, forensic, and vision-language methods. ClueDet achieves the highest operation AP of 34.11 and class-agnostic region AP of 35.13. DINO is the strongest detection baseline at 33.02 operation AP, while Faster R-CNN, RetinaNet, Cascade R-CNN, and RTMDet-M range from 10.55 to 17.07. ClueDet also exceeds WBF and Proposal alignment in operation AP by 4.73 and 0.98 points, respectively. Duplication remains the lowest-AP operation for every operation-aware method, including ClueDet at 13.24 AP.

Among the forensic localizers, DCL-Net and FRD-Net obtain the highest region APs of 13.62 and 13.16. ADCD-Net, Co-Transformers, TruFor, and PIXAR, evaluated using released checkpoints without MEDTD adaptation, obtain between 0.02 and 0.99 region AP after component-to-box conversion. These results characterize performance on MEDTD's operation-involved targets under the stated training and output-conversion settings. Architectural attribution relies on the matched DINO and ClueDet comparisons below rather than the cross-family ranking.

Performance varies substantially within each operation as in Supplementary. For duplication, DINO obtains 0.43 AP on words and 49.71 on pictures. ClueDet improves line-duplication AP from 4.05 to 7.93 and line-splicing AP from 46.13 to 53.24, while word-splicing AP decreases from 11.79 to 10.89. These descriptive results show that aggregate improvements are not uniform across elements.

Ablation study. Table 5 evaluates training coverage and regional aggregation under matched optimization budgets. Increasing distinct training pages from 16,202 to 32,403 raises operation AP from 20.86 to 33.02 for DINO and from 8.68 to 10.55 for Faster R-CNN. Both architectures also improve region AP and reduce FPR at score 0.5. The compared sets cover the same 1,397 recipient documents and all 30 operation–tag combinations, with matched page presentations, optimizer updates, and learning-rate schedules. The comparison therefore contrasts broader page coverage with repeated exposure to a fixed subset, rather than longer training or additional documents.

For regional aggregation, uniform weighting yields 32.56 operation AP, compared with 33.02 for DINO and 34.11 for similarity weighting. The similarity-weighted variant improves over the uniform control by 1.55 AP points, with a paired 95% document-bootstrap interval of [0.93, 2.14]. Both variants retain the same region sampling, masking, projection, insertion layers, and parameter count. The comparison isolates the weighting rule within this module, without changing the training data or supervision. Similarity weighting also achieves 35.13 region AP, compared with 34.00 for uniform aggregation and 34.17 for DINO. However, its FPR of 9.53% remains above DINO's 8.79%, although below the uniform variant's 9.84%. The AP improvement therefore does not imply consistently better false-positive control.

Qualitative results. Figure 3 compares DINO and ClueDet on selected examples of the four operations, displaying all predictions with scores of at least 0.5. Both methods localize the erased region, with scores of 0.62 for DINO and 0.78 for ClueDet. The duplication and splicing examples show additional detections: ClueDet identifies the right-hand duplicated picture with a CD score of 0.56 and the spliced word with a CDS score of 0.58. DINO produces no displayed prediction in either

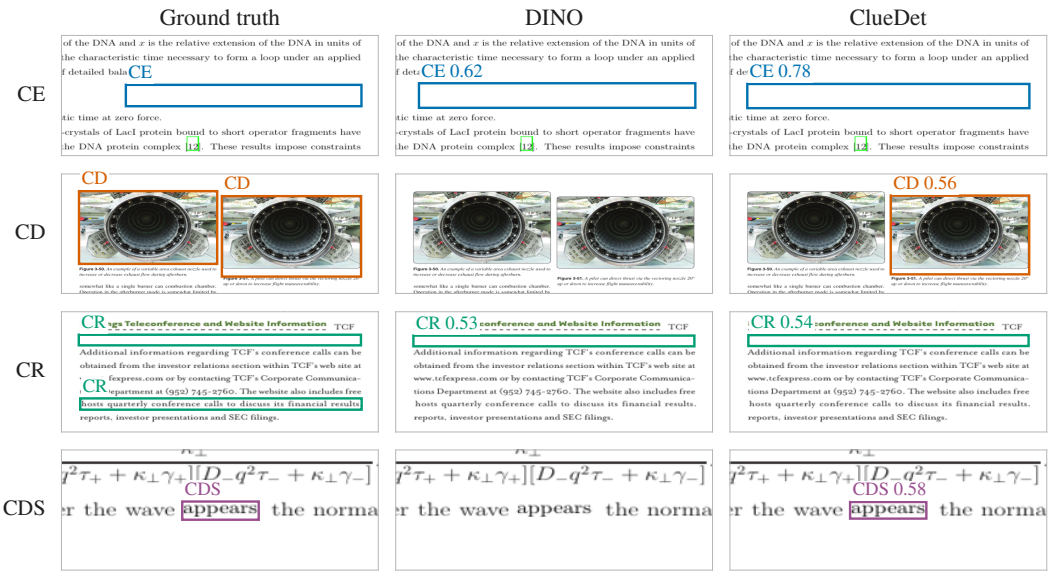


Figure 3: **Qualitative results on MEDTD.** Selected test examples cover the four operations. Labeled boxes show reference regions or predicted classes; numbers are prediction scores. All predictions with score ≥ 0.5 are displayed. Each row uses the same crop across columns, enclosing every reference region and every displayed prediction from both models.

example. These cases illustrate changes in target detection, rather than only confidence differences for regions detected by both models.

Recovery remains incomplete for edits involving two regions. ClueDet detects only one of the two duplication targets. For relocation, both models detect the upper region but miss the lower region, yielding similar predictions despite the additional context module. MEDTD annotates both participating regions in duplication and relocation, so a correctly classified detection may still leave part of an event unlocalized. The examples thus distinguish additional detections from complete recovery of the annotated targets. These observations concern the selected examples at the displayed threshold; omitted regions may receive lower-scoring predictions.

5 Conclusion

We introduced MEDTD, a dataset for detecting editing operations across textual and visual document elements. Full pages, operation-labeled regions, and untampered counterparts enable evaluation of tamper localization, operation recognition, and false-positive behavior. We further introduced ClueDet, which incorporates regional comparison into the decoder to integrate within-page context. The uniform-context ablation shows the benefit of similarity weighting in contextual aggregation. Together, these results highlight the value of within-page contextual comparison for multi-element document tampering detection.

Limitations. The page-coverage comparisons do not isolate coverage from accompanying changes in repetition and target geometry; similarly, comparisons across element types jointly reflect differences in scale, appearance, and frequency. Binary output conversion may merge adjacent targets, while the explicitly marked intermediate checkpoints reflect incomplete training budgets. ClueDet's improvements vary across element types and decision thresholds. Although contextual aggregation improves detection, the experiments do not characterize the regional correspondences learned or used by the model.

ETHICS STATEMENT

MEDTD is intended for research on visual document tampering. Its generation code could also be used to create manipulated documents. A detector identifies visual evidence of editing. Its predictions do not establish intent or factual correctness. Source attribution and modification notices accompany the data. The collection inherits DocLayNet's domain distribution and may contain incidental identifiers in source pages. False-positive results apply to the studied document population and should not be interpreted as validation for a particular deployment.

REPRODUCIBILITY STATEMENT

The main text defines data partitions, detection targets, the contextual update, evaluation metrics, model selection, and uncertainty estimation. Tables and plots are produced from stored evaluation outputs. Data archives, annotations, and the contextual module are available locally. The training settings and output conversions are specified for each comparison. Reported confidence intervals describe document sampling for the fixed trained models.

AI USE STATEMENT

In this work, we used generative AI tools to assist with generating portions of the method implementation code, aggregating selected experimental outputs and interpreting experimental results. We have not used generative AI tools to synthesize dataset images or generate ground-truth annotations, and the development of theoretical proofs is not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final methodological decisions, conclusions, and all content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *Proc. CVPR*, pp. 6154–6162, 2018. doi: 10.1109/CVPR.2018.00644.

Shenghao Chen, Yibo Zhao, Tianyi Wang, Chunjie Ma, Weili Guan, Ming Li, and Zan Gao. Amplifying discrepancies: Exploiting macro and micro inconsistencies for image manipulation localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 35357–35365, 2026. doi: 10.1609/aaai.v40i42.40844.

Yufan Chen, Jiaming Zhang, Kunyu Peng, Junwei Zheng, Ruiping Liu, Philip Torr, and Rainer Stiefelhagen. RoDLA: Benchmarking the robustness of document layout analysis models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Yufan Chen, Ruiping Liu, Junwei Zheng, Di Wen, Kunyu Peng, Jiaming Zhang, and Rainer Stiefelhagen. Graph-based document structure analysis. In *International Conference on Learning Representations*, 2025.

Li Dong, Weipeng Liang, and Rangding Wang. Robust text image tampering localization via forgery traces enhancement and multiscale attention. *IEEE Transactions on Consumer Electronics*, 70(1): 3495–3507, 2024. doi: 10.1109/TCE.2024.3367947.

Bo Du, Xuekang Zhu, Xiaochen Ma, Chenfan Qu, Kaiwen Feng, Zhe Yang, Chi-Man Pun, Jian Liu, and Ji-Zhe Zhou. ForensicHub: A Unified Benchmark & Codebase for All-Domain Fake Image Detection and Localization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. doi: 10.52202/085713-2167. Datasets and Benchmarks Track.

Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. TruFor: Leveraging All-Round Clues for Trustworthy Image Forgery Detection and Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20606–20615, 2023.

Pavel Korshunov, Vidit Vidit, Amir Mohammadi, Christophe Ecabert, Nevena Shamoska, Sébastien Marcel, Zeqin Yu, Ye Tian, Jiangqun Ni, Lazar Lazarevic, Renat Khizbullin, Anastasiia Evteeva, Alexey Tochin, Aleksei Grishin, Anjith George, Daniel Dealcala, Tamas Endrei, Javier Muñoz-Haro, Ruben Tolosana, Ruben Vera-Rodriguez, Aythami Morales, Julian Fierrez, György Cserey, Hardik Sharma, Sachin Chaudhary, Akshay Dudhane, Praful Hambarde, Amit Shukla, Prateek Shaily, Jayant Kumar, Ajinkya Hase, Satish Maurya, Mridul Sharma, and Pallav Dwivedi. DeepID Challenge of Detecting Synthetic Manipulations in ID Documents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 521–530, 2025.

Myung-Joon Kwon, Seung-Hun Nam, In-Jae Yu, Heung-Kyu Lee, and Changick Kim. Learning JPEG compression artifacts for image manipulation detection and localization. *International Journal of Computer Vision*, 130(8):1875–1895, 2022. doi: 10.1007/s11263-022-01617-5.

Weixiang Li, Bin Li, Kengtao Zheng, Songze Li, and Haodong Li. Document image forgery detection and localization in desensitization scenarios. *Signal Processing*, 238:110123, 2026. doi: 10.1016/j.sigpro.2025.110123.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proc. ICCV*, pp. 2980–2988, 2017. doi: 10.1109/ICCV.2017.324.

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(11):7505–7517, 2022. doi: 10.1109/TCSVT.2022.3189545.

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. doi: 10.1016/j.patcog.2024.110828.

Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. RTMDet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*, 2022. doi: 10.48550/arXiv.2212.07784.

Xiaochen Ma, Xuekang Zhu, Lei Su, Bo Du, Zhuohang Jiang, Bingkui Tong, Zeyu Lei, Xinyu Yang, Chi-Man Pun, Jiancheng Lv, and Jizhe Zhou. IMDL-BenCo: A Comprehensive Benchmark and Codebase for Image Manipulation Detection & Localization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024. doi: 10.52202/079017-4277. Datasets and Benchmarks Track.

Soumyaroop Nandi and Prem Natarajan. Can image splicing and copy-move forgery be detected by the same model? Forensim: An attention-based state-space approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6194–6204, 2026.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, Jin Shi, Fan Wu, Pei Chu, Minghao Liu, Zhenxiang Li, Chao Xu, Bo Zhang, Botian Shi, Zhongying Tu, and Conghui He. OmniDocBench: Benchmarking Diverse PDF Document Parsing with Comprehensive Annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24838–24848, 2025.

Jeongsoo Park and Andrew Owens. Community Forensics: Using Thousands of Generators to Train Fake Image Detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8245–8257, 2025.

Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S. Nassar, and Peter W. J. Staar. DocLayNet: A large human-annotated dataset for document-layout segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3743–3751, 2022. doi: 10.1145/3534678.3539043.

Xugong Qin, Jiuqiang Tian, Jiayi Sheng, Tiantian Xia, Yuyi Wang, Chengrui Li, and Gangyan Zeng. Towards fine-grained document tampering detection: New dataset and benchmark. In *Pattern Recognition and Computer Vision: PRCV 2025*, volume 16278, pp. 3–19. Springer, 2026. doi: 10.1007/978-981-95-5676-2_1.

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*, pp. 5937–5946, 2023. doi: 10.1109/CVPR52729.2023.00575.

Chenfan Qu, Yiwu Zhong, Jian Liu, Xuekang Zhu, Bohan Yu, and Lianwen Jin. TextShield-R1: Reinforced reasoning for tampered text detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(10):8621–8629, 2026a. doi: 10.1609/aaai.v40i10.37814.

Chenfan Qu, Yiwu Zhong, Xuekang Zhu, Junchi Li, Changjiang Jiang, Jian Liu, and Lianwen Jin. Detect any AI-counterfeited text image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35437–35450, 2026b.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Xinyi Shang, Yi Tang, Jiacheng Cui, Ahmed Elhagry, Salwa K. Al Khatib, Sondos Mahmoud Bsharat, Jiacheng Liu, Xiaohan Zhao, Jing-Hao Xue, Hao Li, Salman Khan, and Zhiqiang Shen. From masks to pixels and meaning: A new taxonomy, benchmark, and metrics for VLM image tampering. *arXiv:2603.20193*, 2026. doi: 10.48550/arXiv.2603.20193.

Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, 2021. doi: 10.1016/j.imavis.2021.104117.

Yalin Song, Wenbin Jiang, Xiuli Chai, Zhihua Gan, Mengyuan Zhou, and Lei Chen. Cross-attention based two-branch networks for document image forgery localization in the metaverse. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 21(2):1–24, 2024. doi: 10.1145/3686158.

Junke Wang, Zuxuan Wu, Jingjing Chen, Xintong Han, Abhinav Shrivastava, Ser-Nam Lim, and Yu-Gang Jiang. ObjectFormer for Image Manipulation Detection and Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2364–2373, 2022.

Yiming Wang, Baiqi Wu, Qingming Li, Jiahao Chen, Tong Zhang, and Shouling Ji. Order within chaos: Capturing intrinsic energy anomalies for AI-manipulated image forgery localization. *arXiv preprint arXiv:2606.02178*, 2026. doi: 10.48550/arXiv.2606.02178. Accepted to ICML 2026.

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*, pp. 19280–19289, 2025.

Jiaqi Wu, Yuchen Zhou, Muduo Xu, Zisheng Liang, Simiao Ren, Jiayu Xue, Meige Yang, Siying Chen, and Jingheng Huan. AIForge-Doc: A Benchmark for Detecting AI-Forged Tampering in Financial and Form Documents, 2026. Preprint; v1, 2026-02-24.

Muli Yang, Gabriel James Goenawan, Henan Wang, Huaiyuan Qin, Chenghao Xu, Yanhua Yang, Fen Fang, Ying Sun, Joo Hwee Lim, and Hongyuan Zhu. Your AI-Generated Image Detector Can Secretly Achieve SOTA Accuracy, If Calibrated. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14):11622–11630, 2026. doi: 10.1609/aaai.v40i14.38146.

Zeqin Yu, Bin Li, Yuzhen Lin, Jinhua Zeng, and Jishen Zeng. Learning to locate the text forgery in smartphone screenshots. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10095070.

Zeqin Yu, Haotao Xie, Jian Zhang, Jiangqun Ni, Wenkan Su, and Jiwu Huang. Toward real-world text image forgery localization: Structured and interpretable data synthesis. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5085.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *International Conference on Learning Representations*, 2023.

Jiwei Zhang, Wenbo Feng, Siwei Wang, Feifei Kou, Haoyang Yu, and Shaozhang Niu. Collaborative transformers with multi-level forensic attention for image manipulation localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 12556–12563, 2026. doi: 10.1609/aaai.v40i15.38250.

Zengqi Zhao, Weidi Xia, En Wei, Yan Zhang, Jane Mo, Tiannan Zhang, Yuanqin Dai, Zexi Chen, Yiran Tao, and Simiao Ren. DOCFORGE-BENCH: A Comprehensive 0-shot Benchmark for Document Forgery Detection and Analysis, 2026. Preprint; v2, 2026-03-10.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *Proc. ICLR*, 2021. doi: 10.48550/arXiv.2010.04159.

A THE MEDTD DATASET

MEDTD, the Multi-Element Document Tampering Dataset, is designed for detecting editing operations from a single document image. This supplement describes dataset construction and provides the training details, matched comparisons, and element analyses underlying the study of regional context. All table, equation, and section references refer to this document.

A.1 TASK AND PARTITIONS

Task and region targets. Given an RGB page I, a detector predicts a set $\widehat{Y}(I) = \{(\hat{b}_j, \hat{y}_j, \hat{s}_j)\}_{j=1}^{m}$ of bounding boxes, operation labels, and confidence scores. The reference is $Y = \{(b_i, y_i)\}_{i=1}^{n}$, with $Y = \varnothing$ for an untampered page. Each tampered page contains one event. Content Erasure (CE) removes content and annotates its former location. Content Duplication (CD) copies content within a page and annotates both the unchanged source and the inserted copy. Content Relocation (CR) moves content within a page and annotates both the vacated source and the destination. Cross Document Splicing (CDS) inserts content from another page and annotates the receiving region. The source page may belong to the same document. Both targets of a CD or CR event share its operation label. The targets consequently include unchanged or vacated regions involved in an edit, extending beyond the locations of modified pixels.

Inference uses only the input page. The original counterpart, editing operation, target count, ground-truth regions, element labels, and construction metadata are not supplied to the detector.

Page collection and partitions. MEDTD contains programmatically edited, 1025×1025 full-page PNG images derived from DocLayNet (Pfitzmann et al., 2022). Its seven element groups are words, lines, paragraphs, pictures, formulas, tables, and table-cell content. The original annotation tags form 30 observed operation–tag combinations. Section A.2 gives their mapping. Source domains include financial reports, scientific articles, laws and regulations, manuals, patents, and government tenders. The collection comprises 46,290 tampered pages, 25,145 original pages, and 69,839 target boxes from 1,995 source documents.

Partition	Tampered pages	Region boxes	Untampered pages	Documents
Train	32,403	48,887	17,892	1,397
Validation	4,629	6,984	2,404	199
Test	9,258	13,968	4,849	399
Total	46,290	69,839	25,145	1,995

Table 6: MEDTD partitions. Original pages are distinct untampered source images. All original pages and edited derivatives of a recorded recipient document remain in the same partition.

The partitions separate recorded recipient documents within each source collection. Where multiple configurations are compared, validation determines a single configuration that supplies every metric in the corresponding test row.

A.2 DATASET CONSTRUCTION AND ANNOTATIONS

A.2.1 ELEMENT DEFINITIONS AND CANDIDATE EXTRACTION

MEDTD organizes edits by the content of the selected region and its spatial extent. Words, lines, and paragraphs distinguish local text replacement from changes to larger textual units. Pictures, formulas, and tables represent different forms of structured page content. Table-cell content separates local changes within a table from edits to the complete table. These groups describe the construction candidates and evaluation subsets. The detection labels remain CE, CD, CR, and CDS.

The available generation code obtains candidates from text detections and DocLayNet layout annotations. The definitions below specify how each presented element group corresponds to an image region.

Words. The word group uses regions returned by a pretrained docTR DB-ResNet50 text detector. Normalized detections are converted to integer coordinates in the source page. The editing routines operate on the detected image regions.

Lines. Line candidates are enclosing rectangles of grouped word detections. The grouping compares their vertical centers relative to their heights. A word joins a group when its vertical center differs from that of the group's initial word by less than half their mean height. The resulting rectangle spans the minimum and maximum coordinates of the grouped detections. This defines an image-based line candidate without requiring a transcript.

Paragraphs. Paragraph candidates are the DocLayNet regions annotated as Text. This group therefore represents text blocks in the page layout. The annotated text-block extent defines the paragraph candidate.

Pictures. Picture candidates use the complete regions annotated as Picture in DocLayNet. Editing acts on the selected graphical region, providing larger visual structures with boundaries and internal appearance distinct from textual units.

Formulas. Formula candidates use DocLayNet Formula annotations. The corresponding global editing routine operates on the mathematical region as a whole. Its placement code includes formula-specific coordinate adjustments when transferring content to another formula region.

Tables. Table candidates use the complete DocLayNet Table boxes. These edits affect a structured region containing multiple entries and their arrangement, allowing whole-table changes to be examined separately from local tabular edits.

Table-cell content. The local routine selects an annotated table, extracts its image crop, and applies the text detector within that crop. The detected coordinates are then translated back to the complete page. Table-cell content denotes these local text candidates within tabular regions. The candidate extent follows the detected text within the table. This group belongs to the Visual subset through its tabular context even when the selected content consists of text.

A.2.2 EDITING PROCEDURES

Candidate extraction precedes selection of the source and receiving regions. The global layout path expands its annotation boxes by two pixels on each side. Local table extraction skips missing or empty crops. The available editor applies the following operations to candidates in the page image.

Content Erasure. The routine selects one candidate rectangle and fills it with a background color estimated from the page. Sauvola thresholding identifies background pixels, and their most frequent color provides the fill value. The removed region determines the target box.

Content Duplication and Content Relocation. Both routines select two candidates from a group with comparable dimensions. The retained implementation groups candidates whose width and height differ from a reference candidate by less than 10% of the corresponding dimension. It selects the source and receiver from the same group, clears the receiving region, and copies the source content into it using the placement routine. Copy dimensions are limited by the page extent, and incompatible source and destination array shapes are rejected. Duplication retains the source content. Relocation additionally fills the source rectangle with the estimated background. Their annotations retain both participating regions.

Cross Document Splicing. The routine selects another image from the same recorded collection and obtains donor candidates from text detections or layout annotations. A donor candidate is accepted when its width and height each differ from the receiving region by less than 10% of the receiving dimension. Its content is inserted into the recipient page, and the receiving region supplies the localization target. Different images can belong to the same document.

The edited full-page array is saved as PNG before its image and region records are added to the annotation file. Candidate availability, dimensional compatibility, and image-boundary feasibility

determine which events can be constructed on a given page. The final dataset retains one editing event per tampered page.

A.2.3 Source inventory and element distribution

MEDTD starts from DocLayNet page images and their layout annotations. The available entry points enumerate source images through the supplied COCO annotations. Layout-specific routines select pages containing the requested annotated element. Multiple edited derivatives can share a recipient page. The 25,145 original pages are counted by deduplicating the recipient sources represented in the final inventory of 46,290 tampered pages.

Table 7 reports page and target-region counts for the seven semantic groups in each partition. All seven groups contain examples of all four operations in training, validation, and test. Semantic aggregation therefore gives 28 operation–element combinations. The preserved historical tags yield 30 operation–tag combinations, with the mapping described below. Their frequencies reflect the retained examples and available editing candidates. Untampered controls have no editing-event element label and are excluded from this table.

Element	Train		Validation		Test	
	Pages	Regions	Pages	Regions	Pages	Regions
Word	6,209	9,561	847	1,301	2,004	3,038
Line	5,586	8,428	862	1,357	1,627	2,504
Paragraph	5,830	8,691	973	1,502	1,688	2,621
Picture	3,495	5,036	487	680	957	1,425
Formula	3,597	5,118	549	769	1,016	1,374
Table	2,908	4,264	332	437	722	1,004
Table-cell content	4,778	7,789	579	938	1,244	2,002
Total	32,403	48,887	4,629	6,984	9,258	13,968

Table 7: MEDTD element groups by partition. Pages counts tampered pages, each containing one event. Regions counts operation-involved target boxes, including both regions for duplication and relocation. Semantic aliases are merged for this table, while the original operation–tag evaluation retains all 30 observed combinations.

A.2.4 Region targets and element tags

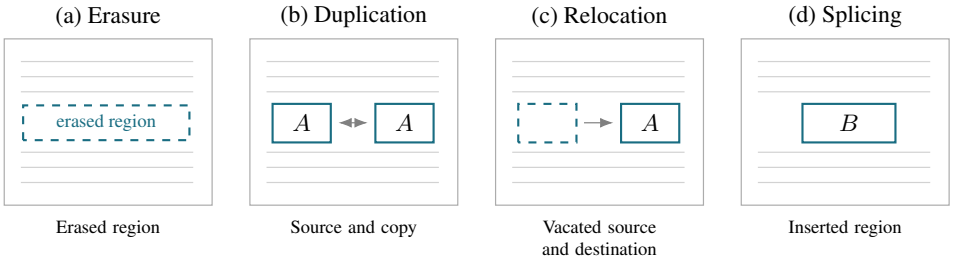


Figure 4: **Region targets for the four operations.** Boxes denote annotated regions. Dashed boxes denote erased or vacated areas. Duplication includes the unchanged source and its copy without assigning a copying direction. Relocation includes the vacated source and destination. The panels illustrate annotation semantics rather than dataset samples.

The original element labels are `word`, `line`, `second_line`, `paragraph`, `picture`, `formula`, `table`, `table_global`, and `table_local`. The seven semantic groups combine `second_line` with line, `table` and `table_global` with table, and `table_local` with table-cell content. Evaluation retains all 30 observed operation–tag combinations under their original labels. The 45 CDS training pages labeled `table` and 752 CR pages labeled `second_line` use alternative names within the same semantic groups.

Each untampered validation page is the source of at least one tampered validation image. Its document identity follows the recorded source-page mapping. Binary-model evaluation converts boxes to a mask by rounding left and top coordinates down, rounding right and bottom coordinates up, clipping to the image boundaries, and taking the union of the resulting half-open rectangles. The mask represents operation-involved regions, including unchanged CD sources and vacated CR sources. It therefore differs from a modified-pixel mask. Untampered pages have no target boxes. Four-class detection uses all original annotations, including both regions of each CD or CR event.

Inspection and source coverage. The authors inspected all 46,290 tampered pages and reinspected them after correction. The retained boxes contain 99.715% of reconstructed changed pixels across the complete dataset. Four duplicate source groups and their eight derivatives were removed from the final inventory. The six source domains contribute 18,770 financial-report, 7,361 scientific-article, 2,794 law-and-regulation, 2,197 manual, 676 patent, and 605 tender pages to training.

Data contents and provenance. The resource contains full-page images, operation-labeled boxes, element tags, recipient-page and recipient-document identifiers, and corresponding original pages. The partitions separate recorded recipient documents. Donor identifiers and a complete record of historical CDS generator versions are unavailable, so donor-document and layout-template independence have not been established. Filenames and metadata can encode editing labels and are excluded from image-only inference. Images and annotations follow CDLA-Permissive-1.0 with upstream attribution, and the accompanying original code uses Apache-2.0. The separate image archive and a complete portable training environment are outside this supplementary document. Anonymous access to the image archive remains to be verified.

B TRAINING AND EVALUATION

Training settings specify the data exposure, optimization budget, and checkpoint used for each MEDTD comparison. Table 8 gives the shared DINO configuration.

Setting	DINO, ClueDet variants, and half-data control
Architecture	Four-scale DINO, ResNet-50; native four-operation head
Initialization	ImageNet ResNet-50 backbone; detector initialization shared across paired models
Training input	Whole-page resize to fit 1333×800 with aspect ratio preserved; no crop/flip
Precision and batch	FP32; one page per microbatch, accumulate two
Optimizer	AdamW, learning rate 10^{-4}, weight decay 10^{-4}
Multipliers and clipping	Backbone learning rate $\times 0.1$; gradient norm clipped at 0.1
Schedule	Linear warmup for 1,000 microsteps from factor 0.001; learning rate $\times 0.1$ after microstep 356,433
Budget	388,836 page presentations; 194,418 optimizer updates
Selection	Seed 0, final fixed checkpoint; no validation during training

Table 8: Shared DINO training configuration for the baseline, context variants, and half-data control. Context variants add the module specified in Section F.1.

Training coverage. The coverage comparison holds optimization length fixed while varying the number of distinct pages. Full-data training makes 12 passes over 32,403 pages. Because each epoch has an odd number of microsteps, gradients accumulate across epoch boundaries. The 388,836-step run ends with a complete update. Half-data training makes 23 complete passes over 16,202 pages, followed by 16,190 pages of the final pass, omitting a 12-page tail. It presents 586,641 target boxes, compared with 586,644 under full-data training. Update counts are matched, but page selection, target geometries, and repetition frequencies differ. Both conditions include all 1,397 training documents and all 30 operation–tag combinations. The comparison uses fixed thresholds, loss weights, checkpoints, and half-data subsets, without searching these choices for the reported result.

Additional full-data detectors. RetinaNet, Cascade R-CNN, and RTMDet-M extend the comparison to additional detection architectures. Each of these three models uses all 32,403 MEDTD tampered training pages and 48,887 boxes for 12 epochs. All use seed 0 and the final checkpoint. RetinaNet and Cascade R-CNN start from ImageNet ResNet-50 backbones with newly initialized detection heads, use batches of two, and make 194,424 updates. SGD uses momentum 0.9, weight decay 10^{-4}, and learning rates 0.00125 and 0.0025, respectively. A 500-update linear warmup precedes tenfold learning-rate reductions after epochs eight and eleven. Whole pages are resized to fit 1333×800 while preserving aspect ratio, without crops or flips.

RTMDet-M starts from the released COCO detector, uses batches of four, and makes 97,212 updates. AdamW uses learning rate 6.25×10^{-5} and weight decay 0.05, with a 1,000-update warmup and cosine decay over the second half of training. Images are resized with aspect ratio preserved and padded to 640×640, retaining all targets. Evaluation uses the final exponential moving average. Each of these three models receives 388,836 page presentations. RetinaNet and Cascade R-CNN make six more optimizer updates than DINO because they use complete per-epoch batches. RTMDet-M uses fewer updates with a batch size of four. The three runs match dataset coverage. Their batch sizes and native training configurations yield different optimizer-step counts and compute costs.

For inference, the three additional detectors use a final score threshold of 0.001 and at most 100 boxes per page. Their native NMS, pre-NMS candidate limits, resizing, and normalization are retained. MMDetection applies a strict comparison at its internal score threshold. The exporter then retains scores of at least 0.001. Input pipelines load only the rendered page and image dimensions. No operation, element, role, or region annotation is provided at inference.

Forensic baselines. The newly adapted forensic baselines use all 32,403 tampered training pages and 4,445 controls selected from the 17,892-page untampered training partition, with an effective batch size of eight. The planned budget is 12 epochs and 55,272 optimizer updates. PSCC-Net, FLAME, and both FRD-Net (Chen et al., 2026) initializations complete this budget. The two FRD-Net initializations are compared by validation class-agnostic AP. The selected checkpoint supplies every reported test metric. CAT-Net v2 (Kwon et al., 2022) uses the training-matched 512×512 image and JPEG representation at quality 100 without chroma subsampling. FLAME retains its mixed-precision forward pass and exponential-moving-average weights. Restored probability maps are stored in FP32. Output conversion is specified in Section F.3.

Incomplete training budgets. The intermediate checkpoints are evaluated before completing their planned training. CAFTB-Net uses 12,000 of 55,272 planned updates. TIFDM uses 27,000 of 55,272 planned updates. CAT-Net v2 uses 14,000 of 55,272 planned updates. Each run uses all tampered training pages. The forensic models additionally use the 4,445 untampered controls defined above. Checkpoints are fixed before fresh validation and test inference and are not selected by test performance.

For each run evaluated before its planned training budget, the available checkpoint is fixed before validation as the sole candidate. No later checkpoint is included in that selection. Every metric in the test row comes from this checkpoint, including the per-operation and mean operation APs of operation-aware detectors. The reported training budget forms part of each experimental condition.

Faster R-CNN coverage control. A second coverage comparison uses Faster R-CNN to assess whether the full-data benefit extends beyond DINO. It uses an ImageNet-initialized ResNet-50 backbone with an FPN and randomly initialized detection heads. Full- and half-data runs share seed 0, detector initialization, aspect-preserving whole-page resizing to 1333×800, and 388,836 microsteps accumulated in pairs. SGD uses learning rate 0.005, momentum 0.9, weight decay 10^{-4}, and gradient clipping at norm 35. Linear warmup spans 1,000 microsteps from factor 0.001. The learning rate decreases tenfold after microsteps 259,224 and 356,433. Both runs retain all targets and use the final iteration. This design tests training coverage within Faster R-CNN.

Binary detector training and released checkpoints. DCL-Net is evaluated with its fixed seed-0 checkpoint from epoch 21. Training uses 36,848 pages, comprising 32,403 tampered and 4,445 untampered pages, with batch size four and 9,212 updates per epoch for 193,452 updates in total. Its

ConvNeXt-Tiny backbone is initialized from ImageNet-22K, with RGB inputs resized to 512×512. The checkpoint is reused without retraining or epoch selection on the present AP results.

The MEDTD comparison reuses this DCL-Net checkpoint on all 9,258 tampered and 4,849 untampered test pages. ADCD-Net, Co-Transformers (Zhang et al., 2026), TruFor, and PIXAR (Shang et al., 2026) use their released checkpoints without MEDTD fine-tuning. ADCD-Net uses the DocTamper checkpoint, combining RGB, frequency, and content features with document restoration. Its MEDTD evaluation obtains PP-OCRv5 detections from the input images. It receives neither original counterparts nor ground-truth operation labels or regions. The evaluated author implementation uses PP-OCRv5 instead of the paper's CRAFT configuration. Co-Transformers and TruFor retain their released inference preprocessing. PIXAR supplies a forced-localization mask.

B.1 Evaluation metrics

Operation AP averages COCO-style average precision over the four classes and ten intersection-over-union (IoU) thresholds $0.50{:}0.05{:}0.95$. Each class is evaluated over all tampered pages. Region AP pools the operation labels and evaluates localization on the same images. AP50 and AP75 fix IoU at 0.50 and 0.75, respectively. AP is reported on a 0–100 scale, with larger values indicating better detection. Operation–tag AP averages the AP of each of the 30 corresponding page subsets, so it differs from global operation AP.

Text AP and Visual AP evaluate region localization on disjoint subsets defined by the edited element. Text comprises words, lines, and paragraphs. Visual comprises pictures, formulas, tables, and table-cell content. The test subsets contain 5,319 and 3,939 tampered pages, respectively. All annotated regions on a selected page remain targets. MEDTD AP uses tampered pages. The false-positive rate (FPR) measures responses on untampered pages separately. Unless stated otherwise, FPR is the fraction of untampered pages with a maximum detection score of at least 0.5, with zero assigned to pages having no exported prediction. Lower FPR is better. Binary localizers use their maximum restored pixel probability as the page score, and their masks are converted to boxes as specified in Section F.3. They do not predict the four operations and therefore have no operation AP.

Four-class detectors export scores of at least 0.001 and at most 100 boxes per page. Localization recall and localized-page TPR additionally require a correctly classified box with IoU of at least 0.5. Section C.1 defines the matching procedure and validation-based threshold selection. Paired 95% intervals for the coverage and context comparisons use 2,000 document-bootstrap samples. The resampling unit is the recorded recipient document.

Selection and statistical estimation. Where alternative configurations are compared, validation selects one complete configuration for every metric in its test row. The coverage and context studies use their fixed final checkpoints. Paired test intervals use 2,000 bootstrap samples of the 399 recipient documents with replacement. Tampered pages and their untampered counterparts remain together in each sample. AP is recomputed on the pooled pages, and paired differences use the same sampled documents. The 2.5th and 97.5th percentiles define the interval. These intervals describe document sampling for fixed seed-0 models and fixed training subsets.

The present MEDTD comparisons use the recipient-document-separated population in Table 6. This test population had been accessed during earlier development. The calibrated-threshold analysis was specified after inspection of partial test aggregates and is exploratory.

C Training coverage and regional context

The comparisons below examine distinct-page coverage and the weighting rule used to aggregate regional context. Table 9 expands the ablations to individual operation classes and operation–tag AP. Table 10 gives paired differences and their uncertainty.

The controlled comparisons separate the effects of training coverage and contextual aggregation. Six final checkpoints are fixed before test inference, comprising full-data DINO, ClueDet, ClueDet (Uniform), half-data DINO, and full- and half-data Faster R-CNN. The test split contains 9,258 tampered pages with 13,968 boxes and 4,849 untampered pages from 399 recipient documents. Each bootstrap replicate samples 399 documents with replacement and pools their pages with the

sampled multiplicities. The same 2,000 replicates, generated with seed 20260915, are used for all models and metrics. All six checkpoints are reported, without selecting models by test performance.

Method	Op. AP	Region AP	Operation–tag AP	FPR	CE	CD	CR	CDS
Full: 32,403 pages								
DINO	33.02	34.17	45.93	8.79	40.93	12.74	35.19	43.21
ClueDet (Uniform) (Ours)	32.56	34.00	45.39	9.84	39.89	11.97	33.94	44.45
ClueDet (Ours)	34.11	35.13	46.84	9.53	41.02	13.24	35.83	46.35
Faster R-CNN	10.55	17.24	25.33	5.65	23.66	2.47	12.75	3.30
Half: 16,202 pages								
DINO	20.86	25.58	30.86	28.81	31.50	5.27	18.19	28.48
Faster R-CNN	8.68	14.55	22.00	16.81	20.05	1.96	10.64	2.09

Table 9: MEDTD test results for the six matched coverage and context models, including operation categories and operation–tag AP. Half denotes the fixed 16,202-page subset and Full denotes all 32,403 tampered training pages. ClueDet (Uniform) uses uniform aggregation.

Test-set comparison	Op. AP	Region AP	FPR (pp)
ClueDet − DINO	+1.09 [+0.49, +1.68]	+0.96 [+0.43, +1.51]	+0.74 [-0.30, +1.82]
ClueDet (Uniform) − DINO	-0.46 [-0.96, +0.10]	-0.16 [-0.65, +0.35]	+1.05 [-0.69, +3.33]
ClueDet − ClueDet (Uniform)	+1.55 [+0.93, +2.14]	+1.12 [+0.47, +1.78]	-0.31 [-2.27, +1.22]
DINO (Full − Half)	+12.16 [+10.79, +13.66]	+8.59 [+7.64, +9.49]	-20.02 [-23.45, -17.10]
Faster R-CNN (Full − Half)	+1.86 [+1.52, +2.27]	+2.70 [+2.13, +3.20]	-11.16 [-12.99, -9.49]
Faster R-CNN − DINO	-22.47 [-24.52, -20.34]	-16.93 [-18.80, -15.08]	-3.13 [-5.17, -1.10]

Table 10: Six prespecified test comparisons. Full and Half denote the training sets in Table 9. The Faster R-CNN–DINO comparison uses Full for both models and is descriptive. Entries are paired differences with 95% document-bootstrap intervals, in AP points or FPR percentage points.

C.1 Detection at calibrated thresholds

This exploratory analysis uses the same six fixed models and the same test population. For each model, the calibration score of an untampered validation page is its maximum exported detection score, or zero if no prediction is exported. The threshold is the next representable value above the descending-score order statistic that allows at most $\lfloor \alpha N \rfloor$ false positives among N validation pages, for $\alpha = 0.01$ or 0.05. Ties are excluded conservatively. A threshold above one suppresses all predictions when necessary. The selected threshold is then applied unchanged to test pages.

Retained predictions are processed by descending confidence, with export order resolving ties. Each prediction is matched to the unmatched same-class target with the highest IoU of at least 0.5. Annotation ID resolves equal IoUs. Recall is the matched fraction of all 13,968 target boxes. Localized-page TPR counts the 9,258 tampered pages with at least one match, and FPR counts the 4,849 untampered pages with any retained prediction. Table 11 distinguishes localization performance from false-positive control at the two validation targets.

Method	Training set	Recall_1	LocTPR_1	FPR_1	Recall_5	LocTPR_5	FPR_5
DINO	Full	8.98	12.54	0.99	17.96	24.92	5.16
ClueDet (Ours)	Full	9.91	13.22	0.82	17.98	24.61	4.48
ClueDet (Uniform) (Ours)	Full	7.35	10.50	0.41	17.44	23.73	4.00
DINO	Half	10.10	14.14	0.66	17.35	24.17	4.39
Faster R-CNN	Full	6.41	9.66	0.85	15.67	23.58	6.17
Faster R-CNN	Half	2.78	4.19	1.07	10.12	15.18	5.16

Table 11: **Localization at validation-calibrated thresholds.** Subscripts 1/5 denote target validation FPR percentages. Recall measures the fraction of ground-truth boxes matched to a same-class prediction at IoU ≥ 0.5. LocTPR is the fraction of tampered pages with at least one match.

D Detection across document elements

Table 12 reports all 30 operation–tag subsets for the six matched models. Each AP is evaluated on its own subset of test pages. Pages and Docs give the number of images and distinct recipient documents, respectively. The Element tag column retains the original annotation labels and aliases, as defined in Section A.2. These descriptive subset values differ from category AP, which uses the complete test population.

The complete breakdown supports comparisons within each editing operation. It includes the word and picture duplication subsets and the line and word splicing subsets discussed in the main analysis. Retaining all subsets exposes variation with element scale, appearance, frequency, and operation, including cases where contextual aggregation decreases AP.

Operation	Element tag	Pages	Docs	DINO (Full)	ClueDet (Ours)	ClueDet (Uniform) (Ours)	DINO (Half)	Faster R-CNN (Full)	Faster R-CNN (Half)
CDS	formula	174	106	60.41	60.27	60.27	43.56	32.25	27.09
CDS	line	377	105	46.13	53.24	49.59	34.57	0.00	0.00
CDS	paragraph	288	127	91.12	90.73	91.67	75.19	12.80	14.52
CDS	picture	88	50	56.03	50.68	54.54	13.26	18.84	18.04
CDS	table	13	13	78.74	91.24	87.59	59.98	63.40	45.40
CDS	table_global	34	22	85.90	83.12	84.88	46.60	48.78	43.48
CDS	table_local	60	36	1.04	1.26	0.87	0.28	0.20	0.16
CDS	word	409	132	11.79	10.89	7.93	2.25	4.68	0.88
CD	formula	140	71	48.84	47.98	47.92	34.14	21.24	19.56
CD	line	449	41	4.05	7.93	4.01	2.03	0.00	0.00
CD	paragraph	460	53	27.43	29.57	26.81	11.46	17.21	14.67
CD	picture	209	69	49.71	49.53	44.36	28.52	41.91	33.95
CD	table_global	152	50	48.10	46.36	43.20	19.17	46.00	29.45
CD	table_local	375	88	2.81	2.33	2.59	1.86	1.19	0.37
CD	word	516	53	0.43	0.33	0.50	0.17	0.12	0.01
CE	formula	484	126	33.13	33.33	32.41	27.01	24.28	20.22
CE	line	373	53	33.91	39.87	36.44	25.47	0.01	0.00
CE	paragraph	467	36	74.69	72.96	73.01	68.64	58.33	56.68
CE	picture	401	143	64.35	64.66	63.73	55.95	52.93	50.00
CE	table_global	393	75	75.19	73.50	73.72	70.62	70.64	69.05
CE	table_local	426	95	36.52	37.00	36.78	27.55	15.17	11.42
CE	word	561	24	41.69	41.48	39.28	31.74	29.74	25.02
CR	formula	218	77	56.47	57.12	54.79	43.50	26.16	23.85
CR	line	178	45	37.85	40.41	39.19	24.55	0.33	0.01
CR	paragraph	473	27	65.61	67.69	64.08	33.19	39.64	35.61
CR	picture	259	75	71.17	73.30	70.38	38.52	51.17	48.16
CR	second_line	250	44	41.52	46.77	41.38	27.28	0.00	0.01
CR	table_global	130	45	71.49	68.93	69.76	38.84	47.72	45.76
CR	table_local	383	85	28.05	28.91	26.34	14.91	10.21	5.82
CR	word	518	65	33.77	33.73	33.59	24.88	24.94	20.75

Table 12: Test AP for all 30 operation–tag combinations and six models. Full and Half denote the training sets in Table 9. Both ClueDet variants use Full. Pages and Docs give test-page and recipient-document counts. These descriptive APs use each combination's page subset rather than the global image population.

E Direct evaluation on external documents

Table 13 reports region localization on fixed collections of 3,444 RTM (Luo et al., 2025) and 3,932 STFD (Yu et al., 2023) tampered images. Both collections contain tampered images only, and the comparison reports region AP. The direct evaluation uses no target fine-tuning. The fixed image collections contain only tampered pages and support region AP, not untampered-page FPR or image-level AUROC.

Method	Checkpoint	RTM AP	STFD AP
DINO	Full MEDTD	0.03	0.16
ClueDet (Ours)	Full MEDTD	0.03	0.23
ClueDet (Uniform) (Ours)	Full MEDTD	0.04	0.20
Co-Transformers	Released	1.72	0.03
ADCD-Net^{o}	Released	2.89	1.03
DCL-Net^{t}	Earlier text-only	0.03	<0.01
CAFTB-Net	Released	1.16	0.06
TIFDM	Released	0.90	0.12
FFDN	Released	5.53	2.24
PSCC-Net	Released	2.11	1.21
FLAME	Released	0.14	<0.01
PIXAR	Released	0.23	0.52

Table 13: **Direct external localization.** Region AP is reported as a percentage on 3,444 RTM and 3,932 STFD tampered images. Every entry belongs to the stated checkpoint and output conversion, with no target fine-tuning. ^{t} denotes the earlier text-only DCL-Net checkpoint, and ^{o} denotes ADCD-Net's RTM run without OCR boxes. These external image collections are evaluated separately from MEDTD.

Region evaluation. The binary baselines' recorded probability maps are quantized to uint8 and thresholded at 128. Eight-connected components of at least four pixels produce boxes scored by mean component probability. COCO evaluation receives all components without NMS and retains at most 100 detections per image. Region AP uses the existing derived box annotations, rather than reconstructing targets from exact pixel masks during evaluation. The resulting box AP values are reported separately for each dataset.

DINO, ClueDet (Uniform), and ClueDet use fixed full-MEDTD checkpoints on these same external images and region annotations, without target training or threshold tuning. Their four native output classes are folded into one region category while preserving box coordinates, scores, and prediction order. On RTM, DINO obtains 0.03 region AP, ClueDet (Uniform) obtains 0.04, and ClueDet obtains 0.03. On STFD, the corresponding values are 0.16, 0.20, and 0.23. Localization accuracy remains low for all three models.

F Implementation details

F.1 Region context

Similarity-weighted and uniform region context. The context ablation isolates how regions are aggregated within the same four-class detector. Both variants use six decoder layers, 900 matching queries, and 256-dimensional features. A shared context block updates the queries immediately *before* decoder layers five and six, corresponding to zero-based indices 4 and 5. Box refinement and the detection head retain the DINO design. The variants have identical parameter counts and use the same dense similarity computation. Only their aggregation weights differ. Table 9 reports detection results, and Section C presents the paired comparisons.

Region features. At each insertion point, let $q_i \in \mathbb{R}^{256}$ denote a query, $b_i = (c_x, c_y, w, h)$ its current normalized reference box, and V_ℓ the encoder features at scale $\ell \in \{1, \ldots, 4\}$. A 3×3 grid with offsets $\{-1/3, 0, 1/3\}^2$ samples bin centers within b_i by bilinear interpolation. Coordinates are clipped to $[0, 1]$ and mapped to the padded feature maps using each scale's valid ratios. Padded

features are zeroed, and interpolation uses `align_corners=False`. The region feature r_i is the mean of the nine samples across all four scales, followed by LayerNorm without trainable parameters. Sampling coordinates are detached from the computation graph. Gradients propagate through encoder features and query representations, but not through reference boxes or the geometric mask.

Region bank and masking. The region bank $R = \{r_j\}_{j=1}^{N}$ contains the region features of the last $N = 900$ matching queries on the same page. The corresponding index set is $\mathcal{B} = \{1, \dots, N\}$. The bank is constructed without confidence or class filtering. The current matching query contributes its own region feature r_i to R. Candidate j is admissible when $j \in \mathcal{B}$ and $\mathrm{IoU}(b_i, b_j) < 0.5$. For a matching query, the additional condition $j \neq i$ excludes the query's own region. Both variants compute this mask from detached boxes, excluding self-matches and pairs with IoU of 0.5 or greater. During training, the bank contains matching-query region features exclusively. Prefix denoising (DN) queries can read this bank. At inference, the decoder contains only matching queries. The original DINO losses and DN procedure retain their training annotations.

For a nonempty admissible set $\mathcal{A}_i$, the aggregation weights are

$$a_{ij}^{\mathrm{sim}} = \frac{\exp(\bar{r}_i^{\mathsf{T}} \bar{r}_j / 0.2)}{\sum_{k \in \mathcal{A}_i} \exp(\bar{r}_i^{\mathsf{T}} \bar{r}_k / 0.2)}, \qquad a_{ij}^{\mathrm{unif}} = \frac{1}{|\mathcal{A}_i|}, \qquad j \in \mathcal{A}_i, \tag{3}$$

where a^{sim} and a^{unif} denote similarity-weighted and uniform aggregation, and $\bar{r}_i = r_i / \max(\|r_i\|_2, 10^{-6})$. Both variants compute dense similarities, multiply them by one for similarity weighting or zero for uniform weighting, and divide by 0.2. Masked logits are set to -10^4. After softmax, masked entries are zeroed and weights are renormalized with a denominator floor of 10^{-12}. For a nonempty set, $c_i = \sum_j a_{ij} r_j$. If $\mathcal{A}_i = \varnothing$, all weights are zero and $c_i = r_i$.

Residual query update. The context block updates each query as

$$\widetilde{q}_i = q_i + f_\theta\big([q_i,\ r_i,\ c_i - r_i]\big), \qquad f_\theta :\ 768 \xrightarrow{\text{Linear}} 256 \xrightarrow{\text{GELU}} 256 \xrightarrow{\text{Linear}} 256. \tag{4}$$

Here, $f_\theta(x) = W_2\,\mathrm{GELU}(W_1 x + b_1) + b_2$, where $W_1 \in \mathbb{R}^{256 \times 768}$ and $W_2 \in \mathbb{R}^{256 \times 256}$. The two linear layers contain $768 \cdot 256 + 256 + 256 \cdot 256 + 256 = 262{,}656$ trainable parameters, including biases, shared across insertion points. Initializing W_2 and b_2 to zero makes the initial residual zero. When no candidate is admissible, the contextual difference $c_i - r_i$ vanishes. The trained feed-forward network (FFN) can nevertheless update the query through its q_i and r_i inputs.

F.2 Output fusion

Output fusion tests whether combining existing predictions yields the gains obtained by contextual query updates. Both baselines use fixed full-data detector checkpoints. For WBF, we combine DINO and Faster R-CNN with equal expert weights. Boxes are clustered separately for each operation. Detection confidence weights the average box coordinates, and the highest member confidence defines the cluster score. This specifies the maximum-confidence variant of WBF. We choose the clustering IoU from $\{0.40, 0.55, 0.70\}$ using validation operation AP. Proposal alignment combines either detector with the fixed epoch-21 DCL-Net probability maps. Forensic proposals are obtained at seven thresholds $\{32, 64, 96, 128, 160, 192, 224\}/255$, with eight-connected components of at least four pixels, nonmaximum suppression at IoU 0.90, and at most 100 proposals per page. The probability maps remain in floating point. We search box weights $\{0, 0.25, 0.50, 0.75\}$, score weights $\{0, 0.25, 0.50\}$, and minimum alignments $\{0.15, 0.30, 0.45\}$. The three equivalent zero-update configurations are treated as one. This gives 34 configurations for each of the two semantic detectors, or 68 in total. Cross-operation competition is disabled. Operation labels are inherited from the detector. The selected rule is applied unchanged to tampered and untampered test pages. Outputs below 0.001 are removed, and at most 100 boxes are retained. Each fusion baseline is selected by unrounded validation operation AP, with the declared candidate order resolving exact ties. The selected settings are IoU 0.70 for WBF, and DINO with box weight 0, score weight 0.50, and minimum alignment 0.45 for Proposal alignment.

F.3 BINARY OUTPUT ADAPTATION

For the binary methods evaluated on MEDTD, float32 probability maps are restored to the original image resolution. Thresholding at $p \geq 0.5$ yields eight-connected components with a minimum area of one pixel. Each component's bounding box is scored by its mean probability. The 100 highest-scoring boxes are retained, with ties resolved by component scan order. Class-agnostic AP uses all original operation-involved target boxes. The conversion produces region predictions from thresholded probability maps. Operation AP is undefined for these methods because they do not predict CE, CD, CR, or CDS. The direct external comparison uses the separate conversion described in Section E.

FPR uses each untampered page's maximum restored pixel probability at threshold 0.5. With the component rules above, the resulting positive-page decisions are identical to thresholding the highest retained component score at 0.5. PIXAR's 100% FPR in the MEDTD comparison belongs to the forced-localization masks. Its separate image classifier labels 653 of the 4,849 untampered pages as real and 4,196 as tampered. Those decisions do not gate the localization output evaluated here.