# ANIMETEXT: A LARGE-SCALE DATASET AND BENCHMARK FOR TEXT DETECTION IN ANIME-STYLE IMAGES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Text in anime-style images includes dialogue, titles, sound effects, signatures, and lettering drawn into the artwork. We introduce *AnimeText*, a dataset of 735,060 images and 4,360,393 text boxes spanning comics, illustrations, posters, and edited images. AnimeText includes four kinds of annotation: fine boxes for compact text regions, coarse boxes for larger groups, polygons for irregular boundaries, and rejected candidates for text-like patterns and poorly localized proposals. We build these annotations with detector proposals and a fine-tuned candidate classifier, and annotators correct the boxes and add text the detector missed. Trained on 5,965 AnimeText images, YOLO11m reaches 0.850 F1 on the full test split of 73,725 images, compared with 0.545 when trained on the same number of Manga109s images. In a labeled sample of test images, non-dialogue text such as sound effects, titles, and signatures accounts for most of the newly detected regions, and its recall rises from 18.3% to 69.8%; dialogue recall also rises from 63.4% to 91.3%. Training on 513,159 AnimeText images raises F1 from 0.850 to 0.896 and $AP_{50:95}$ from 0.781 to 0.882. Very small text and crowded pages remain the hardest cases.

## 1 INTRODUCTION

Text is part of both the language and the visual composition of anime-style images. Dialogue appears inside comic panels, titles are integrated into illustrations, signatures identify their creators, and expressive lettering depicts sound or motion. A text detector must find these regions amid the surrounding artwork and determine their spatial extent. Cropping, masking, and editing text in these images all start from accurate detection.

Existing resources each cover part of this range. Natural-scene benchmarks annotate incidental, multilingual, and arbitrarily shaped text in photographs (Karatzas et al., 2015; Ch'ng & Chan, 2017; Nayef et al., 2019; Singh et al., 2021). Manga109 and later manga datasets annotate text on scanned comic pages (Fujimoto et al., 2016; Aizawa et al., 2020; Del Gobbo & Matuk Herrera, 2020), and COO adds artistic sound effects (Baek et al., 2022). These collections center on photographs or comic pages. We ask whether a training resource that also covers illustrations, posters, and edited images, and annotates sound effects, titles, and signatures alongside dialogue, improves detection with the same number of training images, and which kinds of text account for the gain. Such a resource needs annotations for compact lettering, larger text groups, irregular boundaries, and patterns that resemble text.

We build AnimeText for this purpose. It contains 735,060 images with 4,360,393 accepted text boxes and 565,497 rejected candidates, ranging from sparse illustrations to dense pages and from horizontal to vertical and curved text of widely varying size. Printed, handwritten, and decorative lettering appears in dialogue, narration, titles, sound effects, and attribution alike.

Four annotation forms describe where text is and how it groups (Figure 1): fine boxes enclose compact text regions, coarse boxes enclose groups of neighboring fine boxes, polygons trace irregular boundaries, and rejected candidates mark text-like patterns and proposals that cut through text or include too much background.



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.

With the same 5,965 training images, YOLO11m trained on AnimeText reaches 0.850 F1 on the full test split, compared with 0.545 after Manga109s training, and recall rises by 35.7 points. In a labeled sample of 256 test images, non-dialogue text accounts for 63.0% of the newly detected regions, and its recall rises from 18.3% to 69.8%; dialogue recall also rises by 27.9 points. Enlarging the AnimeText training set to 513,159 images raises F1 from 0.850 to 0.896 and $AP_{50:95}$ from 0.781 to 0.882. AnimeText training also improves DBNet and LRANet, and weighting rejected candidates in the loss raises $AP_{50:95}$ from 0.701 to 0.806. Very small text and crowded pages remain difficult.

Our contributions are as follows.

- **Dataset.** AnimeText covers comics, illustrations, posters, and edited images, and annotates dialogue, sound effects, titles, signatures, and in-scene text.
- **Annotations.** Fine boxes, coarse boxes, polygons, and rejected candidates describe the extent, grouping, and boundaries of text and the patterns that resemble it.
- **Benchmark.** On 73,725 test images, AnimeText training outperforms Manga109s training with the same number of images, mostly by detecting non-dialogue text, and keeps improving with more data.

## 2 Related Resources

**Natural-scene and structured text.** ICDAR15 targets incidental scene text, while Total-Text, CTW-1500, and ArT focus on curved or arbitrarily shaped text (Karatzas et al., 2015; Ch'ng & Chan, 2017; Liu et al., 2019; Chng et al., 2019). MLT-2019 adds multilingual text (Nayef et al., 2019), TextOCR annotates text in natural images at scale for detection, recognition, and visual reasoning (Singh et al., 2021), and HierText labels words, lines, and paragraphs (Long et al., 2022).

**Manga, comics, and artistic lettering.** Manga109 annotates text and layout on scanned comic pages (Fujimoto et al., 2016; Aizawa et al., 2020). Del Gobbo & Matuk Herrera (2020) study manga text detection with semi-automatic pixel-level annotation, COO targets arbitrary or truncated onomatopoeia (Baek et al., 2022), and MangaSeg adds pixel-level semantic and instance segmentation to Manga109 (Xie et al., 2025).

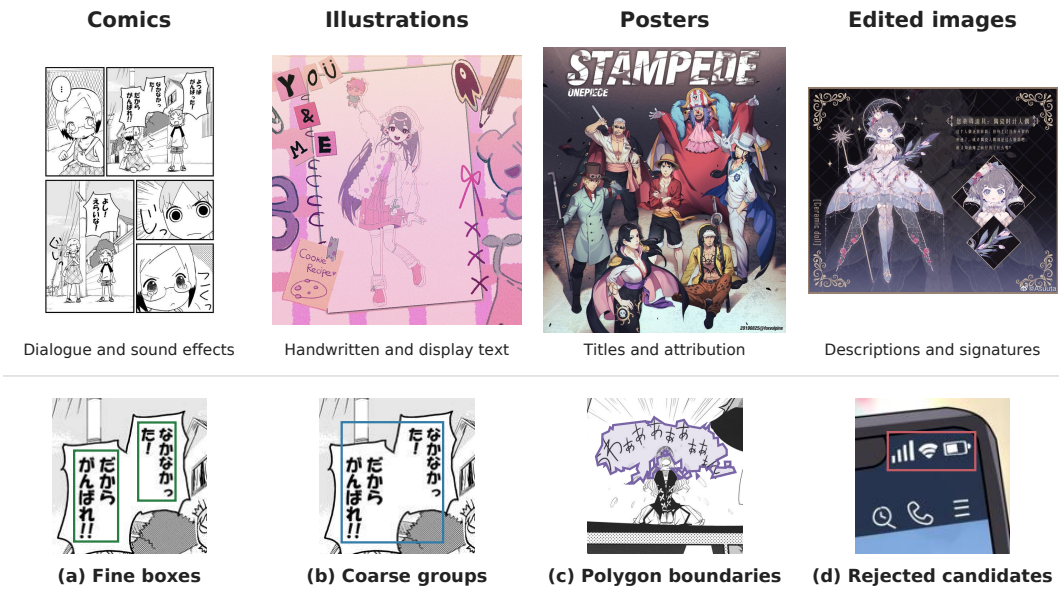



Figure 1: Content and annotations in AnimeText. Top: text in comics, illustrations, posters, and edited images serves as dialogue, sound effects, titles, and attribution. Bottom: the four annotation forms.

Table 1: Scale, text coverage, and localization units of related resources. Counts follow each resource's image unit, including two-page manga spreads. Manga109s is the Manga109 collection used in our experiments; CoMix is an evaluation benchmark.

| Resource | Images | Image/text coverage | Localization units |
|---|---|---|---|
| ICDAR15 | 1,500 | Incidental natural-scene text | Word quadrilaterals |
| CTW-1500 | 1,500 | Curved natural-scene text | Text-line polygons |
| TextOCR | 28,134 | Text in natural images | Word regions |
| HierText | 11,639 | Natural scenes and documents | Words, lines, paragraphs |
| Manga109s | 8,519 | Scanned manga; text regions | Native text boxes |
| COO | 10,602 | Manga; artistic sound effects | Onomatopoeia polygons, links |
| MangaSeg | 10,130 | Manga109; text and layout | Text-instance masks |
| CoMix | ∼3,800 | Cross-style comics; dialogue | Text boxes for evaluation |
| AnimeText | 735,060 | Comics, illustrations, posters, edited images | Fine/coarse boxes, polygons, rejected candidates |




Figure 2: Printed, handwritten, and decorative text in English, Russian, Chinese, Japanese, and Korean, followed by patterns that resemble text.

CoMix evaluates multi-task comic understanding across artistic styles, including detection and dialogue association (Vivoli et al., 2024); its detection evaluation excludes onomatopoeia and scene text unrelated to dialogue. Compared with these resources (Table 1), AnimeText adds standalone illustrations, posters, and edited images to comics and annotates signatures, watermarks, and display text alongside dialogue, over 735,060 images.

## 3 The AnimeText Dataset

### 3.1 Content and Scale

AnimeText contains anime-style images collected online, including illustrations, comics, screenshots, edited images, posters, fan works, and AI-generated images. The text is in English, Chinese, Japanese, Korean, and Russian, often combined with visual effects or decorative elements. It can be central to a page, drawn on an object, or placed near the border as a title or attribution (Figure 2).

The dataset is split into 514,144 training, 147,191 validation, and 73,725 test images (Table 2). Training and test images come from different sources, so the test split measures generalization beyond the training sources. The 4,360,393 accepted text boxes include both fine and coarse boxes, and 565,497 further boxes mark rejected candidates. Polygons cover 720,934 images with 5,685,319 text components and 780,053 rejected-candidate components.

### 3.2 Dataset Construction

**Seed annotation and proposal expansion.** Figure 3 outlines the pipeline. Annotators labeled about 50,000 text-containing images as the seed set, and separate reviewers checked and corrected these labels. A YOLO11 detector (Jocher & Qiu, 2024) trained on the seed set then proposed text regions in the remaining images. Annotators inspected the images, corrected inaccurate boxes, and added text the detector missed, such as stylized artistic lettering and low-contrast text (Figure 4, left).

Table 2: Image and box counts in AnimeText. Accepted boxes include fine and coarse boxes.

| Split | Images | Accepted text boxes | Rejected candidates |
|---|---|---|---|
| Train | 514,144 | 3,028,826 | 430,454 |
| Validation | 147,191 | 867,168 | 123,208 |
| Test | 73,725 | 464,399 | 11,835 |
| Total | 735,060 | 4,360,393 | 565,497 |

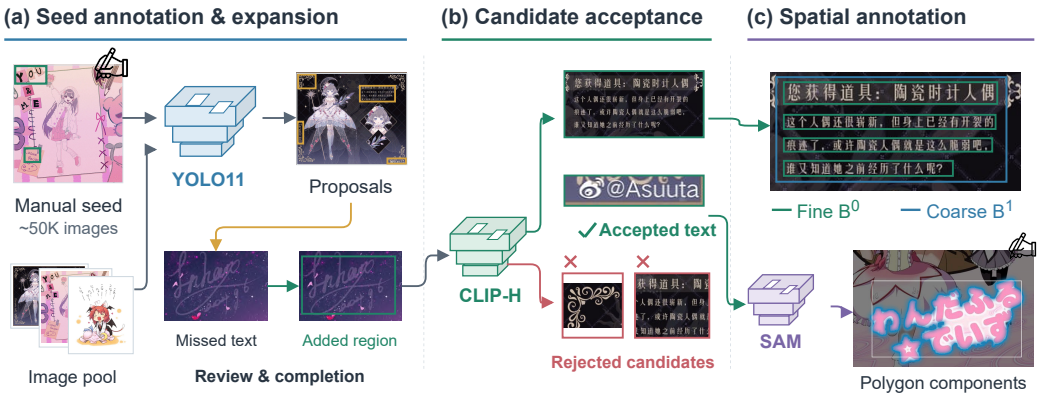



Figure 3: Construction of AnimeText. (a) A manually annotated seed set trains YOLO11 to propose text regions; annotators correct the boxes and add missed text. (b) A fine-tuned CLIP-H classifier separates accepted regions from rejected candidates. (c) Accepted regions are organized into fine and coarse boxes, and box-prompted SAM produces polygons that annotators refine.

**Candidate acceptance.** We fine-tuned a CLIP-H image encoder (Radford et al., 2021; Cherti et al., 2023) on manually labeled candidate crops to separate accepted text regions from rejected ones. The classifier rejects text-like patterns, including pseudo-text in generated images, as well as crops that truncate text or include too much background (Figure 4, right). We keep the rejected boxes as an annotation category; Section A gives the training details.

**Boundary annotation.** SAM (Kirillov et al., 2023) generated polygon proposals from box prompts, and annotators refined inaccurate ones. A region can consist of several polygon components, for example when expressive lettering is split into separate parts. Boxes and polygons thus annotate the same text regions.

### 3.3 Annotation Forms

**Fine boxes.** A fine box $B^0$ tightly encloses a row, column, or compact group $C$ of characters. Character proximity is the gap between the closest edges of character boxes, with threshold

$$\tau_0(C) = \alpha \min\big(\bar{w}(C), \bar{h}(C)\big), \qquad \alpha \approx 0.5, \tag{1}$$

where $\bar{w}(C)$ and $\bar{h}(C)$ are the mean character width and height. Each character in a group lies within $\tau_0$ of another group member, and characters outside the group lie farther away (Section A). A row or column separated from its neighbors forms its own fine box, even within the same paragraph. Our detection benchmark evaluates fine boxes.

**Coarse boxes.** A coarse box $B^1$ encloses a group $G = \{B_i^0\}$ of neighboring fine boxes, grouped with the threshold

$$\tau_1(G) = \beta \min\big(\bar{w}(G), \bar{h}(G)\big), \tag{2}$$

where the mean dimensions are taken over the fine boxes in $G$ and $\beta$ sets the coarse spacing. Each coarse box keeps its constituent fine boxes (Section B).

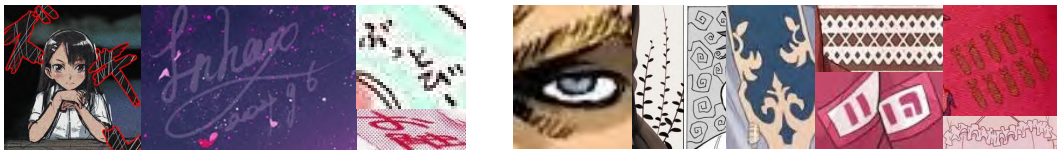



Figure 4: Examples from dataset construction. Left: missing samples, text missed by the proposal detector and added by annotators. Right: hard negative samples, that is, rejected candidates such as text-like patterns and poorly localized proposals.

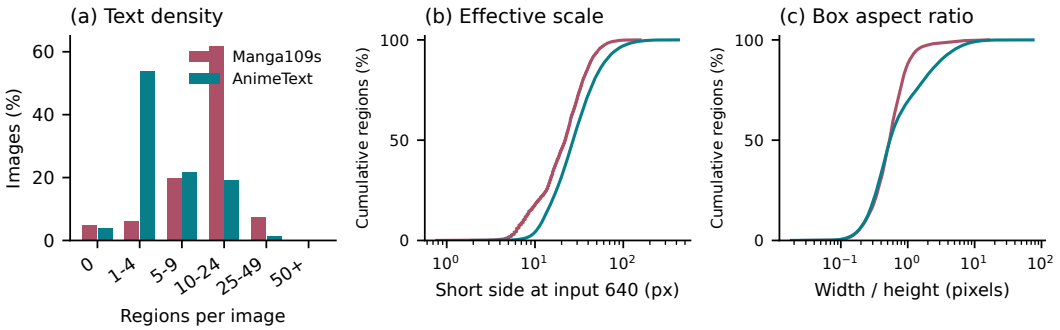



Figure 5: Geometry of AnimeText fine boxes and Manga109s text boxes in the test splits. (a) Text regions per image. (b) Box short side after resizing the longest image side to 640 pixels. (c) Box aspect ratio (width/height). Curves in (b) and (c) are cumulative distributions.

**Polygon boundaries.** A text region with polygon components $P_1, \ldots, P_k$ is represented by their union $M = \bigcup_{j=1}^{k} P_j$. The polygons enclose the characters together with the background between them, follow irregular boundaries, and keep separated parts of a region as distinct components.

**Rejected candidates.** Rejected candidates are text-like patterns that are not text, and text proposals with the wrong extent. A well-localized artistic word, watermark, or signature counts as accepted text, whereas a crop that truncates text or includes too much background is rejected. Extent is judged at the intended granularity, so the spacing between fine boxes inside a coarse box is acceptable. Section 4.6 uses these candidates in training.

### 3.4 Text Density, Scale, and Shape

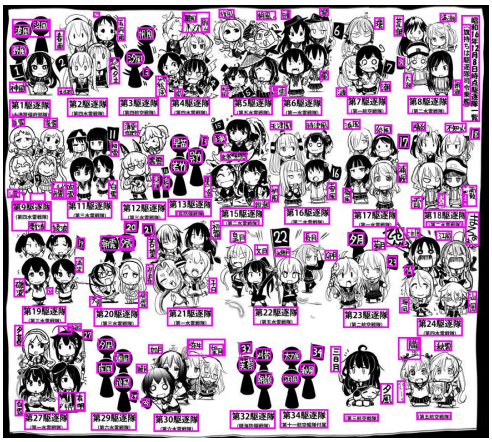
Figure 5 compares fine boxes in the 73,725 AnimeText test images with text boxes in the 862 Manga109s test images. AnimeText images have a median of 3 text regions, compared with 13 for Manga109s, but they range from a single region to 43 images with at least 50 regions (Figure 6). A detector must therefore both find isolated text within artwork and separate many nearby regions.

At a 640-pixel input, the median box short side is 27.0 pixels in AnimeText and 22.1 in Manga109s. Median aspect ratios are similar (0.54 and 0.53), but the 95th percentiles are 4.87 and 1.38: AnimeText adds a long tail of wide horizontal text to compact and vertical text. Section D reports area and location distributions, and Section 4.7 relates density and size to detection accuracy.

### 3.5 Text Functions and Visual Forms

We sample 256 text-containing test images uniformly at random and label all 1,423 fine boxes in them with Qwen3.8-27B (Qwen Team, 2026), which sees the full image and numbered crops. Each region receives three independent labels: function, lettering, and layout (Section C.1).

Dialogue accounts for 50.9% of the regions (Figure 7); the other 698 (49.1%) include 307 sound effects, 98 texts on depicted objects, 90 titles or display texts, and 66 credits, signatures, or watermarks. Of the sampled images, 79 contain text but no dialogue.



(a) Repeated panels and small labels.



(b) Mixed lettering in an illustrated page.



(c) Detail from (a).




(d) Detail from (b).

Figure 6: Dense text layouts in AnimeText. Magenta boxes are annotated text regions; (c) and (d) magnify parts of (a) and (b).

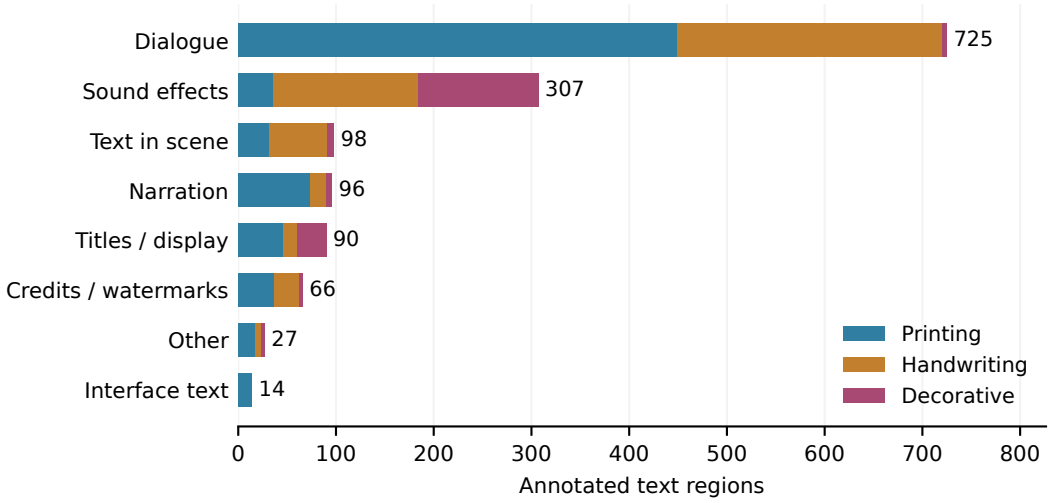



Figure 7: Function and lettering of 1,423 fine text regions in 256 sampled test images, labeled with Qwen3.8-27B. Bars group regions by function; colors show lettering.

Printed text accounts for 49.5% of the regions, handwriting for 37.9%, and decorative lettering for 12.6%. Lettering does not follow function: sound effects include 148 handwritten and 123 decorative regions, and dialogue includes 271 handwritten ones. Vertical text makes up 60.2%, horizontal text 35.4%, and curved or rotated text 4.1%. Section 4.3 uses these labels to locate the detection gains.

Table 3: YOLO11m trained on different resources and data sizes and evaluated on fine boxes in all 73,725 test images. The first two rows use the same number of training images.

| Resource | Train images | Precision | Recall | F1 | $AP_{50}$ | $AP_{50:95}$ |
|---|---|---|---|---|---|---|
| Manga109s | 5,965 | 0.6795 | 0.4553 | 0.5453 | 0.4541 | 0.2433 |
| AnimeText | 5,965 | 0.8926 | 0.8120 | 0.8504 | 0.9034 | 0.7811 |
| AnimeText | 35,000 | 0.9054 | 0.8535 | 0.8787 | 0.9353 | 0.8441 |
| AnimeText | 513,159 | 0.9255 | 0.8680 | 0.8958 | 0.9579 | 0.8822 |

Table 4: Recall (%) and its change (points) by text function on the 256-image sample for the two 5,965-image models, with gained and lost regions.

| Function | Regions | Manga109s | AnimeText | $\Delta$ recall | Gained | Lost |
|---|---|---|---|---|---|---|
| Dialogue | 725 | 63.4 | 91.3 | +27.9 | 215 | 13 |
| Narration | 96 | 40.6 | 82.3 | +41.7 | 42 | 2 |
| Sound effects | 307 | 18.9 | 72.3 | +53.4 | 167 | 3 |
| Titles / display | 90 | 11.1 | 62.2 | +51.1 | 47 | 1 |
| Attribution | 66 | 7.6 | 57.6 | +50.0 | 33 | 0 |
| In-scene text | 98 | 12.2 | 67.3 | +55.1 | 55 | 1 |
| Interface text | 14 | 7.1 | 57.1 | +50.0 | 7 | 0 |
| Other | 27 | 11.1 | 66.7 | +55.6 | 15 | 0 |
| All text | 1423 | 41.3 | 80.7 | +39.4 | 581 | 20 |
| Non-dialogue | 698 | 18.3 | 69.8 | +51.4 | 366 | 7 |

# 4 Benchmark and Experiments

## 4.1 Experimental Setup

Our benchmark evaluates fine-box detection on all 73,725 test images, which contain 422,417 fine boxes. The main study trains YOLO11m (Jocher & Qiu, 2024) on 5,965 images from Manga109s or from AnimeText, and on nested AnimeText subsets of 35,000 and 513,159 images. All models start from the same COCO-pretrained weights and train for 50 epochs with batch size 128, AdamW at learning rate $10^{-3}$, and 640-pixel inputs; we evaluate the final checkpoint (Section C).

Precision, recall, and F1 use one-to-one matching at IoU 0.5, with each model's confidence threshold chosen to maximize F1 on a held-out set of 5,000 validation images. $AP_{50}$ and $AP_{50:95}$ are computed over the full test split. The candidate-supervision study (Section 4.6) and the cross-resource comparisons in Section 4.5 use an all-box validation protocol: they evaluate all text boxes, including coarse boxes, on the validation split at fixed confidence thresholds (Table 7).

## 4.2 Training Resource Comparison

Table 3 reports the main study. With the same 5,965 training images, AnimeText training raises F1 from 0.545 to 0.850 and $AP_{50}$ from 0.454 to 0.903, even though the AnimeText subset contains fewer than half as many text regions (32,759 versus 78,704). Precision rises from 0.679 to 0.893 and recall from 0.455 to 0.812.

## 4.3 Detection Gains Across Text Types

Most of the gain comes from text outside dialogue. On the 256-image sample of Section 3.5, the AnimeText-trained model detects 581 regions that the Manga109s-trained model misses, while missing only 20 that it finds (Figure 9). Non-dialogue text makes up 63.0% of the gained regions but only 49.1% of all regions (Figure 8).

Non-dialogue recall rises from 18.3% to 69.8%, and dialogue recall from 63.4% to 91.3%. The gain spans every non-dialogue function (Table 4): recall rises from 18.9% to 72.3% for sound effects, from 12.2% to 67.3% for in-scene text, and from 7.6% to 57.6% for credits, signatures, and watermarks.

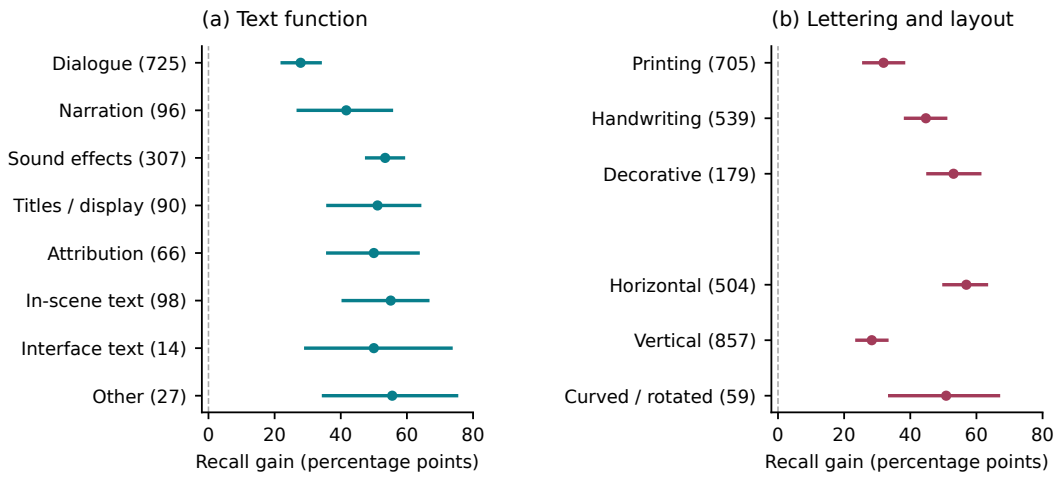



Figure 8: Recall gain of AnimeText over Manga109s training (5,965 training images each) on the 256-image sample, in percentage points. Lines show 95% paired bootstrap intervals over test images; parentheses give region counts.

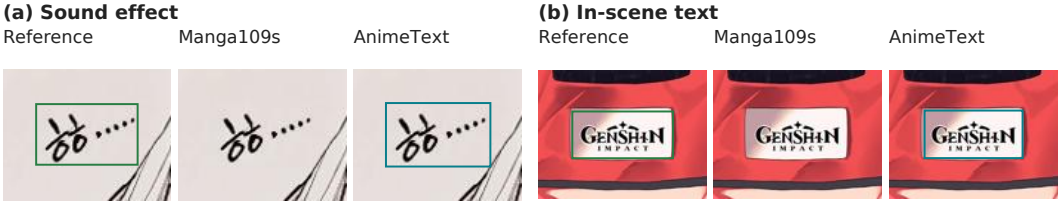



Figure 9: Detections after training on 5,965 Manga109s or AnimeText images: (a) a handwritten sound effect and (b) text on an illustrated object. Green: reference; orange: Manga109s; teal: AnimeText. More examples appear in Section E.

The gains are smallest where the Manga109s-trained model is already strongest: vertical text, typical of Japanese manga dialogue, rises from 61.5% to 89.8%, and printed text from 55.0% to 87.0%. They are largest where that model rarely detects text: horizontal text rises from 11.7% to 68.7%, decorative lettering from 9.5% to 62.6%, and curved or rotated text from 3.4% to 54.2%. Across the sample, recall rises by 39.4 points (95% bootstrap interval [35.0, 43.6]).

### 4.4 Training Scale

More AnimeText data keeps improving detection (Table 3). From 5,965 to 35,000 to 513,159 training images, F1 rises from 0.850 to 0.879 to 0.896 and $AP_{50:95}$ from 0.781 to 0.844 to 0.882, with precision and recall both improving at each step. $AP_{50}$ rises by 5.45 points while $AP_{50:95}$ rises by 10.11, so the gains grow at stricter IoU thresholds: additional data also tightens the predicted boxes.

### 4.5 Other Detectors and Cross-Resource Transfer

The benefit carries over to text-specific detectors (Table 5). Trained on AnimeText instead of ICDAR15, DBNet (Liao et al., 2020) reaches 0.743 F1 and LRANet (Su et al., 2024) 0.855, with both precision and recall improving.

AnimeText training also transfers to manga: a YOLO11m trained only on AnimeText reaches 0.870 F1 on Manga109s, compared with 0.939 for a model trained on Manga109s itself. Section F reports all cross-resource results.

Table 5: DBNet and LRANet trained on ICDAR15 or AnimeText and evaluated on AnimeText under the all-box validation protocol.

| Detector | Training resource | Precision | Recall | F1 |
|---|---|---|---|---|
| DBNet | ICDAR15 | 0.022 | 0.005 | 0.008 |
| DBNet | AnimeText | 0.851 | 0.660 | 0.743 |
| LRANet | ICDAR15 | 0.124 | 0.145 | 0.134 |
| LRANet | AnimeText | 0.833 | 0.878 | 0.855 |

Table 6: Training with rejected candidates. YOLO11m is evaluated on all text boxes of the validation split; $AP_{50:95}$ is computed for the text class.

| Training treatment | Precision | Recall | $AP_{50:95}$ |
|---|---|---|---|
| All candidates | 0.668 | 0.766 | 0.701 |
| Candidate filtering | 0.857 | 0.819 | 0.756 |
| All candidates + loss weighting | 0.878 | 0.825 | 0.806 |

### 4.6 Candidate Filtering and Loss Weighting

Rejected candidates are most useful when they enter the loss with their own weight (Table 6). We compare three ways of training on candidates: keeping text and rejected candidates as two classes (all candidates), treating rejected candidates as background (candidate filtering), and keeping them while scaling the whole loss of each rejected candidate by a fixed weight $\lambda_{\mathrm{rej}}$ that differs from the background weight $\lambda_{\mathrm{bg}}$ (loss weighting). $AP_{50:95}$ on text rises from 0.701 with all candidates to 0.756 with filtering and 0.806 with loss weighting, which also gives the highest precision (0.878) and recall (0.825).

### 4.7 Remaining Localization Challenges

Crowded pages and very small text remain the hardest cases, although more data helps both. On the 43 test images with at least 50 fine boxes, recall rises from 56.1% to 64.9% and 68.5% across the three AnimeText training sizes, but stays below the 90.7% reached on images with 10–24 boxes. For the 5,656 boxes whose short side is below eight pixels at a 640-pixel input, recall rises from 44.6% to 57.5% and 59.5%, compared with 88.2% and 89.8% for boxes of 8–16 and 16–32 pixels.

On the 2,820 test images without text, the full-scale model produces 0.445 false positives per image, fewer than the 0.560 of the 5,965-image model; rejecting text-like background remains a target for future detectors (Section D.1).

## 5 Conclusion

AnimeText makes it possible to train and evaluate text detectors on anime-style images beyond comic pages. With the same number of training images, it teaches a detector to find sound effects, titles, signatures, and in-scene text that Manga109s training largely misses, while also improving dialogue detection. More AnimeText data continues to raise accuracy, most visibly at strict IoU thresholds, and AnimeText training also improves DBNet and LRANet. Its fine boxes, coarse boxes, polygons, and rejected candidates provide supervision for box detection, grouping, boundary prediction, and candidate rejection on the same images. Detecting very small text, separating crowded regions, and suppressing false detections on text-free images are the clearest targets for future detectors.

## AI Use Statement

AI tools assisted with English editing and software development. Qwen3.8-27B labeled the text content of sampled regions and helped select qualitative examples. The authors are responsible for the manuscript, analysis, and empirical claims.

## REFERENCES


Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset "Manga109" with annotations for multimedia applications. *IEEE MultiMedia*, 27(2):8–18, 2020.

Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. COO: Comic onomatopoeia dataset for recognizing arbitrary or truncated texts. In *European Conference on Computer Vision*, 2022.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2818–2829, 2023.

Chee Kheng Ch'ng and Chee Seng Chan. Total-Text: A comprehensive dataset for scene text detection and recognition. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pp. 935–942. IEEE, 2017.

Chee Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, et al. ICDAR2019 robust reading challenge on arbitrary-shaped text (RRC-ArT). In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1571–1576. IEEE, 2019.

Julián Del Gobbo and Rosana Matuk Herrera. Unconstrained text detection in manga: a new dataset and baseline. *arXiv preprint arXiv:2009.04042*, 2020.

Azuma Fujimoto, Toru Ogawa, Kazuyoshi Yamamoto, Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Manga109 dataset and creation of metadata. In *Proceedings of the 1st International Workshop on Comics Analysis, Processing and Understanding*, pp. 1–5, 2016.

Glenn Jocher and Jing Qiu. Ultralytics YOLO11. `https://github.com/ultralytics/ultralytics`, 2024.

Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman K. Ghosh, Andrew D. Bagdanov, Masakazu Iwamura, Jiri Matas, Lukás Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, Faisal Shafait, Seiichi Uchida, and Ernest Valveny. ICDAR 2015 competition on robust reading. In *13th International Conference on Document Analysis and Recognition, ICDAR 2015, Nancy, France, August 23-26, 2015*, pp. 1156–1160, 2015.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11474–11481, 2020. doi: 10.1609/aaai.v34i07.6812.

Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90:337–345, 2019.

Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1049–1059, 2022.

Nibal Nayef, Yash Patel, Michal Busta, Pinaki Nath Chowdhury, Dimosthenis Karatzas, Wafa Khlif, Jiri Matas, Umapada Pal, Jean-Christophe Burie, Cheng-lin Liu, et al. ICDAR2019 robust reading challenge on multi-lingual scene text detection and recognition (RRC-MLT-2019). In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1582–1587. IEEE, 2019.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork. `https://qwen.ai/blog?id=qwen3.8`, August 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, volume 139, pp. 8748–8763, 2021.

Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. TextOCR: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8802–8812, 2021.

Yuchen Su, Zhineng Chen, Zhiwen Shao, Yuning Du, Zhilong Ji, Jinfeng Bai, Yong Zhou, and Yu-Gang Jiang. LRANet: Towards accurate and efficient scene text detection with low-rank approximation network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4979–4987, 2024.

Yunjie Tian, Qixiang Ye, and David Doermann. YOLOv12: Attention-centric real-time object detectors. *arXiv preprint arXiv:2502.12524*, 2025.

Emanuele Vivoli, Marco Bertini, and Dimosthenis Karatzas. CoMix: A comprehensive benchmark for multi-task comic understanding. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Minshan Xie, Jian Lin, Hanyuan Liu, Chengze Li, and Tien-Tsin Wong. Advancing manga analysis: Comprehensive segmentation annotations for the Manga109 dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

## A Construction and Annotation Guidelines

**Seed annotation and proposal expansion.** The seed set contained about 50,000 images, with separate annotators for initial labeling and for review and correction. The proposal detector was YOLO11 (Jocher & Qiu, 2024) trained on the seed set for 50 epochs with Adam at learning rate $10^{-3}$, and we kept proposals with confidence of at least 0.45. Annotators then corrected inaccurate boxes and added missed text, much of it stylized or low-contrast lettering (Figure 4).

**Candidate classification.** The CLIP-H classifier (OpenCLIP ViT-H/14 trained on LAION-2B) was fine-tuned on manually labeled text and rejected patches. Fine-tuning adapts the upper visual-encoder blocks, the projection, and a cosine classification head whose weights are initialized from the text embeddings of `[text, pattern]`. The classifier rejects non-text patterns, including pseudo-text in generated images, and text proposals with the wrong extent.

**Fine and coarse grouping.** For a set $S$ of character or fine boxes, the mean dimensions are

$$\bar{w}(S) = \frac{1}{|S|} \sum_{u \in S} w(u), \qquad \bar{h}(S) = \frac{1}{|S|} \sum_{u \in S} h(u). \tag{3}$$

Let $d(u, v)$ be the gap between the closest edges of boxes $u$ and $v$, and let $\mathcal{U}$ be the set of all elements of the same granularity in the image. A group with at least two elements satisfies

$$\begin{aligned} &\forall u \in S,\ \exists v \in S \setminus \{u\} : \quad d(u, v) \le \tau(S), \\ &\forall u \in S,\ \forall v \in \mathcal{U} \setminus S : \quad d(u, v) > \tau(S). \end{aligned} \tag{4}$$

Fine groups use character boxes with $\tau(S) = \tau_0(S)$ (Equation (1)), and a single character can form its own fine box. Annotators apply this rule to rows, columns, and stylized layouts and draw a tight box around each group, so a row or column separated from neighboring text forms its own $B^0$. Coarse groups use fine boxes with $\tau(S) = \tau_1(S)$ (Equation (2)), and each $B^1$ keeps its fine boxes.

**Rejected candidates.** Rejected candidates are of two kinds. A semantic negative is a non-text pattern or symbol that resembles text. A localization negative contains text but truncates it or includes too much background, for example a crop through the middle of a word; the spacing between the fine boxes of a coarse box does not make it a localization negative.

## B Spatial Annotation Representations

Fine and coarse levels follow from box containment (Figure 10). A box is the parent of another if it is larger and covers at least 95% of the other's area; within a containment chain, the smallest such box is the parent. Fine boxes are the leaves, and coarse boxes are parents of at least two fine boxes. Each polygon component belongs to the smallest box of the same category that covers at least 95% of its area, and the components of a box form the region $M$ of Section 3.3.



(a) Spatial grouping. (b) Boundary geometry.

Figure 10: Spatial annotations. (a) Coarse boxes group nearby fine boxes. (b) Polygons follow irregular text regions.

## C Main YOLO11m Study

Table 7 lists the evaluation protocol of each experiment group.

Table 7: Evaluation protocols on AnimeText. All text boxes include fine and coarse boxes; the threshold rule applies to precision and recall.

| Experiment group | Evaluation targets | P/R threshold |
|---|---|---|
| Main YOLO11m study | Test split; fine boxes | Selected on validation |
| All-box experiments (YOLO11m, DBNet, LRANet) | Validation split; all text boxes | Fixed threshold |
| Released YOLO12 detectors | Validation split; all text boxes | Fixed threshold |

**Training data.** The Manga109s training split contains 5,965 images with 78,704 text regions; we use its text category as the detection target. The AnimeText subsets of 5,965, 35,000, and 513,159 images are nested and drawn with a fixed hash ranking. After removing 985 training images that also appear in the validation or test split, 513,159 of the 514,144 training images remain.

**Training settings.** All four models are YOLO11m (20.05M parameters) initialized from the same COCO-pretrained weights and trained for 50 epochs with batch size 128, AdamW (learning rate $10^{-3}$, weight decay $5 \times 10^{-4}$), a three-epoch warmup, and a linear decay to 0.01 of the initial rate. Mosaic augmentation is turned off for the last ten epochs, and flips are not used. Table 8 lists the training data and compute.

Table 8: Training data and compute for the main YOLO11m study. Hours are training time on one H200; the last column is the confidence threshold selected on validation.

| Resource | Images | Text regions | Batches | Hours | Threshold |
|---|---|---|---|---|---|
| Manga109s | 5,965 | 78,704 | 2,350 | 0.30 | 0.02 |
| AnimeText | 5,965 | 32,759 | 2,350 | 0.30 | 0.48 |
| AnimeText | 35,000 | 186,968 | 13,700 | 1.68 | 0.44 |
| AnimeText | 513,159 | 2,728,745 | 200,500 | 24.59 | 0.47 |

**Evaluation.** The test target contains all 73,725 test images and their 422,417 fine boxes (Section B), including 2,820 images without text. Thresholds are selected on 5,000 validation images (27,094 fine boxes) by maximizing F1 over confidences from 0.01 to 0.95 in steps of 0.01. Inference uses 640-pixel inputs, NMS IoU 0.7, and at most 300 detections per image. The content analysis and qualitative examples use the two 5,965-image models at the same thresholds.

Table 9: Detection at a common confidence threshold of 0.25 on the full test split and on the 256-image content sample.

| Resource | Train images | Precision | Recall | F1 |
|---|---|---|---|---|
| Complete test split | | | | |
| Manga109s | 5,965 | 0.809 | 0.374 | 0.511 |
| AnimeText | 5,965 | 0.841 | 0.872 | 0.856 |
| AnimeText | 35,000 | 0.867 | 0.906 | 0.886 |
| AnimeText | 513,159 | 0.882 | 0.932 | 0.906 |
| Content sample | | | | |
| Manga109s | 5,965 | 0.813 | 0.333 | 0.473 |
| AnimeText | 5,965 | 0.862 | 0.864 | 0.863 |

Table 9 reports the same models at a common threshold of 0.25. The Manga109s-trained model reaches $AP_{50} = 0.958$ on the Manga109s validation split (1,692 images).

Table 10: The two 5,965-image models on the 256-image content sample.

| Training resource | Precision | Recall | F1 | $AP_{50}$ | $AP_{50:95}$ |
|---|---|---|---|---|---|
| Manga109s | 0.680 | 0.413 | 0.514 | 0.422 | 0.219 |
| AnimeText | 0.905 | 0.807 | 0.854 | 0.909 | 0.787 |

### C.1 Text-Content Categories and Paired Gains

Each region receives independent function, lettering, and layout labels. Function categories are dialogue or thought, narration, onomatopoeia, titles and display text, credits/signatures/watermarks, text on objects in the scene, interface text, and other; mixed and uncertain labels cover the rest.

Lettering categories are printing, handwriting or calligraphy, and decorative lettering. Bold or colored printing counts as printing, while deformed shapes, graphic textures, or illustration-like strokes make lettering decorative. A handwritten signature is thus a credit by function and handwriting by lettering, and decorative dialogue is still dialogue. Layout categories are horizontal, vertical, curved or rotated, mixed, and uncertain, judged from the visible arrangement of the text.

**Classification prompt.** Each request gives Qwen3.8-27B the full image and up to six numbered crops of reference regions, and asks for one function, lettering, and layout label per crop in JSON.

Category recall is the fraction of reference regions in a category that a model detects. A gained region is detected only by the AnimeText-trained model; a lost region, only by the other.

**Content-specific results.** Table 10 gives overall results on the sample, and Tables 4 and 11 give recall and gained and lost regions for every category. The intervals in Figure 8 come from 5,000 paired bootstrap resamples of the 256 images, recomputing category recall on each resample.

Table 11: Recall (%) and its change (points) by lettering and layout for the two 5,965-image models, with gained and lost regions.

| Attribute | Regions | Manga109s | AnimeText | Δ recall | Gained | Lost |
|---|---|---|---|---|---|---|
| Lettering | | | | | | |
| Printing | 705 | 55.0 | 87.0 | +31.9 | 232 | 7 |
| Handwriting | 539 | 34.0 | 78.7 | +44.7 | 252 | 11 |
| Decorative | 179 | 9.5 | 62.6 | +53.1 | 97 | 2 |
| Layout | | | | | | |
| Horizontal | 504 | 11.7 | 68.7 | +56.9 | 297 | 10 |
| Vertical | 857 | 61.5 | 89.8 | +28.4 | 252 | 9 |
| Curved / rotated | 59 | 3.4 | 54.2 | +50.8 | 31 | 1 |
| Mixed | 3 | 0.0 | 33.3 | +33.3 | 1 | 0 |

# D GEOMETRY AND FULL-TEST DETECTION

We compare the 422,417 fine boxes in the 73,725 AnimeText test images with the 11,477 text boxes in the 862 Manga109s test images (Table 12 and Figures 11 and 12). Relative area is box area over image area, aspect ratio is width over height, and the short side is measured after resizing the longest image side to 640 pixels.

Table 12: Geometry of AnimeText fine boxes and Manga109s text boxes in the test splits.

| Statistic | Manga109s | AnimeText |
|---|---|---|
| Images | 862 | 73,725 |
| Text regions | 11,477 | 422,417 |
| Mean regions per image | 13.31 | 5.73 |
| Median regions per image | 13 | 3 |
| Median relative box area (%) | 0.329 | 0.703 |
| Median short side at input 640 (pixels) | 22.1 | 27.0 |
| Median pixel aspect ratio | 0.53 | 0.54 |
| 95th-percentile pixel aspect ratio | 1.38 | 4.87 |

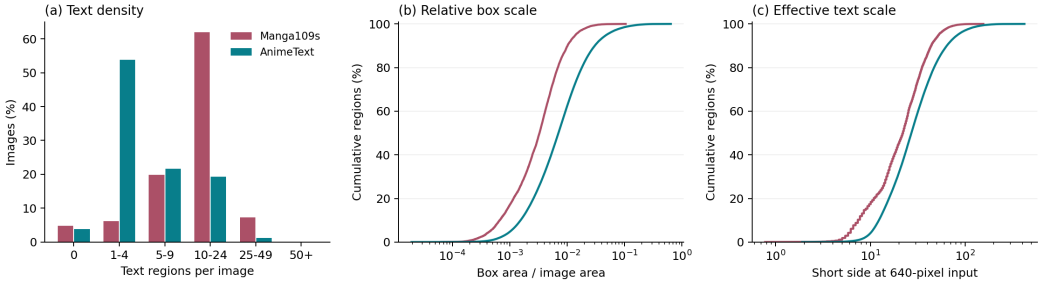



Figure 11: Text regions per image, relative box area, and box short side at a 640-pixel input. The area and size panels are cumulative distributions.

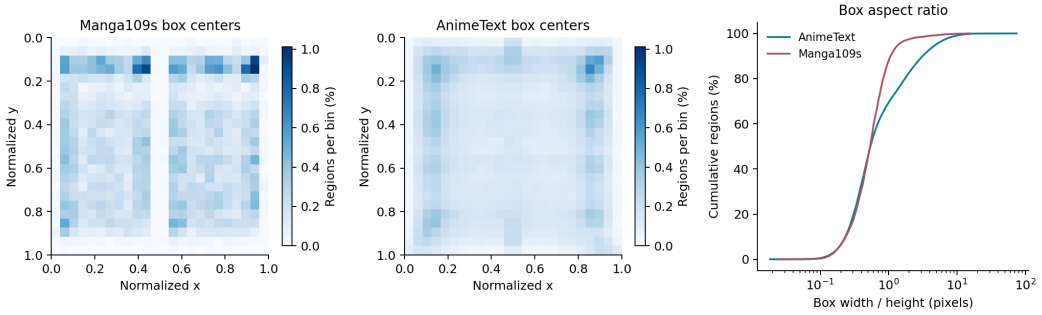



Figure 12: Box-center locations (each 24×24 heatmap normalized to 100%) and box aspect ratios. Both resources place text near image borders, and AnimeText has a longer tail of wide boxes.

## D.1 FULL-TEST ERROR STRATA

Tables 13 and 14 break down the test recall of the four main-study models by the number of fine boxes per image and by box short side at a 640-pixel input.

Table 13: Test recall (%) by fine boxes per image.

| | | Manga109s | AnimeText | | |
|---|---|---|---|---|---|
| Regions/image | Images | 5,965 | 5,965 | 35,000 | 513,159 |
| 1–4 | 39,703 | 27.7 | 71.1 | 76.4 | 77.3 |
| 5–9 | 15,967 | 45.8 | 81.8 | 85.5 | 87.0 |
| 10–24 | 14,251 | 53.5 | 85.5 | 89.2 | 90.7 |
| 25–49 | 941 | 44.4 | 81.2 | 86.4 | 88.2 |
| 50+ | 43 | 13.7 | 56.1 | 64.9 | 68.5 |

Table 14: Test recall (%) by box short side at a 640-pixel input.

| | | Manga109s | AnimeText | | |
|---|---|---|---|---|---|
| Short side (pixels) | Regions | 5,965 | 5,965 | 35,000 | 513,159 |
| $< 8$ | 5,656 | 22.3 | 44.6 | 57.5 | 59.5 |
| $[8, 16)$ | 82,719 | 47.9 | 81.9 | 86.5 | 88.2 |
| $[16, 32)$ | 166,954 | 47.8 | 84.8 | 88.4 | 89.8 |
| $\geq 32$ | 167,088 | 42.9 | 78.5 | 82.6 | 84.0 |

Table 15 reports false detections on the 2,820 test images without text. Training on all AnimeText data reduces them from 1,579 to 1,255 compared with the 5,965-image model.

Table 15: False detections on the 2,820 test images without text. The last column counts images with at least one false detection.

| Resource | Train images | False positives | FP/image | Images with FP |
|---|---|---|---|---|
| Manga109s | 5,965 | 1,130 | 0.401 | 497 |
| AnimeText | 5,965 | 1,579 | 0.560 | 942 |
| AnimeText | 35,000 | 1,614 | 0.572 | 1,004 |
| AnimeText | 513,159 | 1,255 | 0.445 | 822 |

Recall improves with AnimeText training size in every density and size bin, including from 64.9% to 68.5% on the 43 most crowded images between 35,000 and 513,159 training images. The full-scale model reaches 59.5% recall below eight pixels and 68.5% on the most crowded images, compared with 89.8% for 16–32 pixel boxes and 90.7% for images with 10–24 boxes.

## E Paired Qualitative Examples

We selected six images from the content sample, covering several text functions and one regression, from 36 candidates ranked by Qwen3.8-27B using the category labels and the gained and lost regions. Each row shows the image with the focal region highlighted, followed by the same crop with the reference boxes and with each model's predictions.

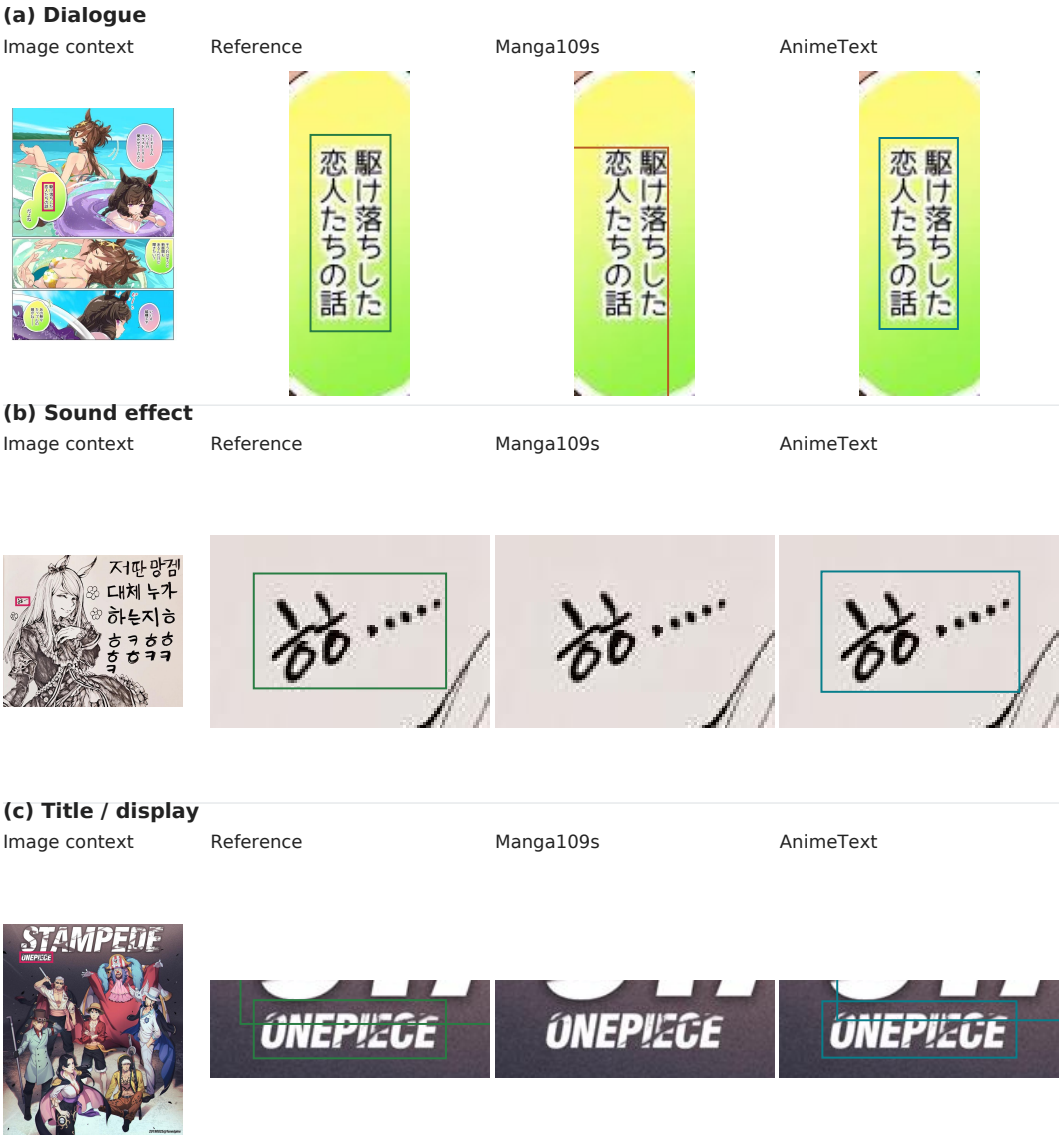
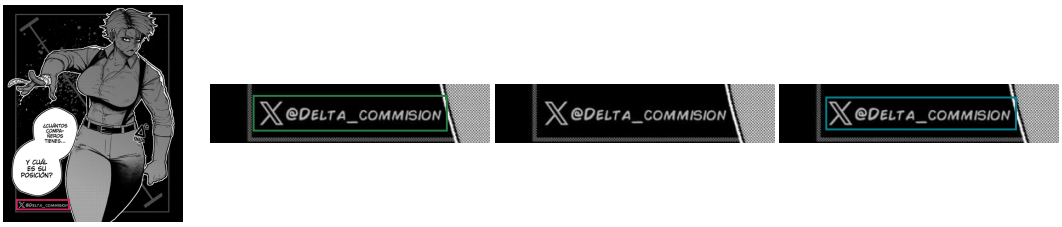



Figure 13: Paired examples: (a) vertical dialogue, (b) a handwritten sound effect, and (c) title text. Green: reference; orange: Manga109s training; teal: AnimeText training (5,965 images each).

**(d) Attribution**

Image context | Reference | Manga109s | AnimeText




**(e) In-scene text**

Image context | Reference | Manga109s | AnimeText




**(f) Regression**

Image context | Reference | Manga109s | AnimeText

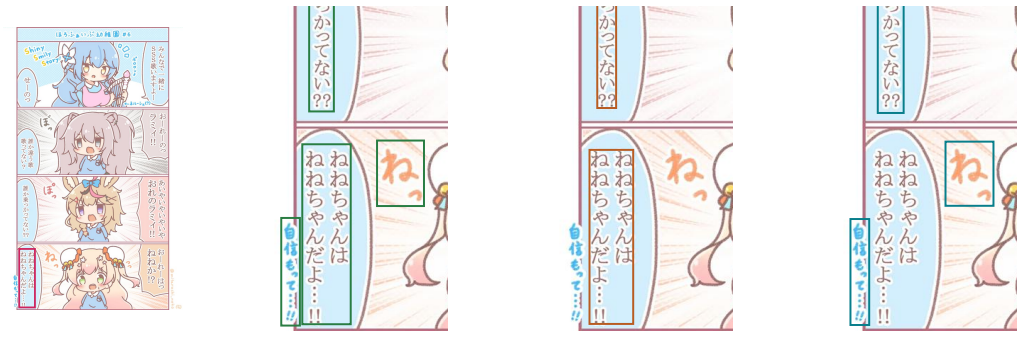



Figure 14: More paired examples: (d) attribution, (e) in-scene text, and (f) a regression, in which the AnimeText-trained model detects nearby lettering but misses a dialogue region found by the Manga109s-trained model. Colors follow Figure 13.

# F YOLO11m Experiments with All-Box Validation

The YOLO11m models in these experiments, including the candidate-supervision study in Table 6, are trained for 50 epochs with batch size 512 and AdamW at learning rate $10^{-3}$. These experiments and the DBNet and LRANet comparisons evaluate AnimeText on all text boxes of the validation split at fixed confidence thresholds (Table 7). For cross-resource evaluation, ICDAR15 words are grouped into text blocks.

## F.1 Cross-Resource Detection

Table 16 compares training resources on AnimeText. YOLO11m trained on AnimeText reaches 0.851 F1, compared with 0.079, 0.2493, and 0.5455 when trained on ICDAR15, CTW-1500, and Manga109s. Manga109s training reaches 0.8646 precision but only 0.3985 recall, whereas AnimeText training reaches 0.825 recall at 0.878 precision.

Table 16: Training resources compared on AnimeText under the all-box validation protocol. AP denotes $AP_{50:95}$; DBNet and LRANet rows repeat Table 5.

| Detector | Training resource | Precision | Recall | F1 | AP |
|---|---|---|---|---|---|
| YOLO11m | ICDAR15 | 0.0629 | 0.106 | 0.079 | 0.010 |
| YOLO11m | CTW-1500 | 0.2335 | 0.2673 | 0.2493 | 0.0698 |
| YOLO11m | Manga109s | 0.8646 | 0.3985 | 0.5455 | 0.3742 |
| YOLO11m | AnimeText | 0.878 | 0.825 | 0.851 | 0.806 |
| DBNet | ICDAR15 | 0.022 | 0.005 | 0.008 | – |
| DBNet | AnimeText | 0.851 | 0.660 | 0.743 | – |
| LRANet | ICDAR15 | 0.124 | 0.145 | 0.134 | – |
| LRANet | AnimeText | 0.833 | 0.878 | 0.855 | – |

Table 17 summarizes cross-resource F1, and Table 18 gives full in-domain and cross-resource results, each evaluated against the target resource's own text annotations.

Table 17: YOLO11m cross-resource F1. Rows: training resource; columns: evaluation resource.

| Training resource | ICDAR15 | CTW-1500 | Manga109s | AnimeText |
|---|---|---|---|---|
| ICDAR15 | 0.565 | 0.4597 | – | 0.079 |
| CTW-1500 | 0.4106 | 0.7986 | – | 0.2493 |
| Manga109s | – | – | 0.9394 | 0.5455 |
| AnimeText | 0.233 | 0.3666 | 0.8696 | 0.851 |

Table 18: YOLO11m in-domain and cross-resource results.

| Training resource | Evaluation resource | Precision | Recall | F1 | $AP_{50:95}$ |
|---|---|---|---|---|---|
| ICDAR15 | ICDAR15 | 0.657 | 0.495 | 0.565 | 0.291 |
| AnimeText | ICDAR15 | 0.187 | 0.310 | 0.233 | 0.083 |
| CTW-1500 | CTW-1500 | 0.8072 | 0.7901 | 0.7986 | 0.5503 |
| AnimeText | CTW-1500 | 0.4224 | 0.3238 | 0.3666 | 0.1754 |
| CTW-1500 | ICDAR15 | 0.4579 | 0.3722 | 0.4106 | 0.1661 |
| ICDAR15 | CTW-1500 | 0.5120 | 0.4171 | 0.4597 | 0.2394 |
| Manga109s | Manga109s | 0.9453 | 0.9336 | 0.9394 | 0.8215 |
| AnimeText | Manga109s | 0.8441 | 0.8966 | 0.8696 | 0.6912 |

## F.2 Training Scale with All-Box Validation

Under this protocol, enlarging the YOLO11m training set from 35,000 to 514,144 images raises F1 from 0.743 to 0.851 and $AP_{50:95}$ from 0.712 to 0.806 (Table 19), gains of 10.8 and 9.4 points.

Table 19: YOLO11m training scale under the all-box validation protocol.

| Training images | Precision | Recall | F1 | $AP_{50:95}$ |
|---|---|---|---|---|
| 35,000 | 0.751 | 0.736 | 0.743 | 0.712 |
| 514,144 | 0.878 | 0.825 | 0.851 | 0.806 |

## G RELEASED YOLO12 DETECTORS

We release five YOLO12 (Tian et al., 2025) detectors trained on AnimeText, from 2.57M to 59.1M parameters, in checkpoint and ONNX formats. Table 20 reports their results under the all-box validation protocol, and Section G.1 analyzes YOLO12l on the content sample.

Table 20: Released YOLO12 detectors under the all-box validation protocol.

| Model | Params (M) | Precision | Recall | $AP_{50}$ | $AP_{50:95}$ |
|---|---|---|---|---|---|
| YOLO12n | 2.57 | 0.86490 | 0.85504 | 0.92310 | 0.83449 |
| YOLO12s | 9.25 | 0.87922 | 0.87581 | 0.93891 | 0.86523 |
| YOLO12m | 20.1 | 0.88725 | 0.88466 | 0.94556 | 0.87804 |
| YOLO12l | 26.4 | 0.89415 | 0.89060 | 0.94905 | 0.88587 |
| YOLO12x | 59.1 | 0.89785 | 0.89292 | 0.95167 | 0.89656 |

### G.1 YOLO12L ON THE CONTENT SAMPLE

On the 256-image content sample of Section 3.5, the released YOLO12l detects 1356 of the 1,423 fine boxes (95.3% recall) at confidence 0.25 and IoU 0.5; Table 21 breaks this down by category.

Recall is 97.2% for printed, 94.2% for handwritten, and 91.1% for decorative lettering, and 97.3% for vertical compared with 88.1% for curved or rotated text.

Dialogue reaches 97.8% recall, compared with 85.7% for in-scene text, 91.1% for titles and display text, and 90.9% for attribution (Figure 15). Non-dialogue text accounts for 51 of the 67 misses (76.1%) while making up 49.1% of the regions.

**Spatial scale.** By box short side at a 640-pixel input (Table 22), YOLO12l detects 10 of the 14 boxes below eight pixels and 95.1–96.2% of larger boxes; 27 of its 67 misses have a short side of at least 32 pixels.

Table 22: YOLO12l recall by box short side at a 640-pixel input on the content sample.

| Short side (pixels) | Regions | Detected | Recall (%) |
|---|---|---|---|
| $< 8$ | 14 | 10 | 71.4 |
| $[8, 16)$ | 327 | 311 | 95.1 |
| $[16, 32)$ | 527 | 507 | 96.2 |
| $\geq 32$ | 555 | 528 | 95.1 |

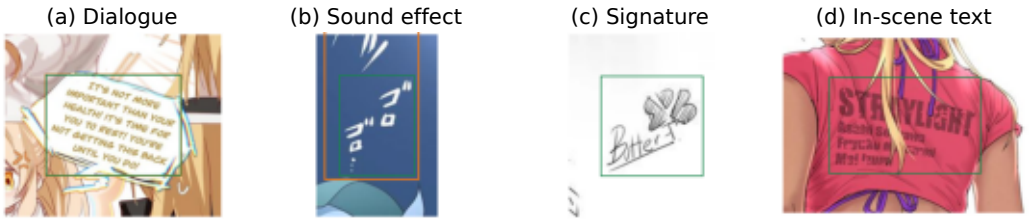



Figure 15: YOLO12l examples. Green: reference; orange: the overlapping prediction with the highest IoU. (a) Matched dialogue; (b) wrong extent; (c, d) missed signature and in-scene text.

Table 21: YOLO12l recall by text category on the 256-image content sample.

| Category | Regions | Detected | Recall (%) |
|---|---|---|---|
| Text function | | | |
| Dialogue | 725 | 709 | 97.8 |
| Narration | 96 | 92 | 95.8 |
| Sound effects | 307 | 289 | 94.1 |
| Titles / display | 90 | 82 | 91.1 |
| Credits / watermarks | 66 | 60 | 90.9 |
| Text in scene | 98 | 84 | 85.7 |
| Interface text | 14 | 14 | 100.0 |
| Other | 27 | 26 | 96.3 |
| Lettering | | | |
| Printing | 705 | 685 | 97.2 |
| Handwriting | 539 | 508 | 94.2 |
| Decorative | 179 | 163 | 91.1 |
| Layout | | | |
| Horizontal | 504 | 467 | 92.7 |
| Vertical | 857 | 834 | 97.3 |
| Curved / rotated | 59 | 52 | 88.1 |
| Mixed | 3 | 3 | 100.0 |