Supplementary Material: MIST: Multilingual Incidental Dataset for Scene Text Detection

Saumya Mundra
IIIT Hyderabad, India
saumyamundra@gmail.com

Ajoy Mondal
IIIT Hyderabad, India
ajoy.mondal@iiit.ac.in

C.V Jawahar
IIIT Hyderabad, India
jawahar@iiit.ac.in

Contents

A. Annotation Guideline

The key annotation guidelines provided to annotators are as follows:

- Bounding box annotations follow the COCO-Text approach, where a word is defined as an uninterrupted sequence of characters separated by spaces.
- Word-level annotations use polygons, similar to Total-Text and CTW1500. Given the dataset's multilingual and multi-oriented text, polygonal bounding boxes ensure precise annotations.
- All human-legible text must be tightly annotated, regardless of challenging conditions such as poor illumination or motion blur. If a text instance appears illegible at first glance, annotators should adjust contrast or brightness to determine its readability.
- Occluded words should be annotated as a whole rather than in segments.
- We also annotate the illegible text and mark it as *do not care* during evaluation.
- Personally Identifiable Information (PII) must be annotated and labeled - PII. During post-processing, these marked areas are blurred to protect sensitive data.

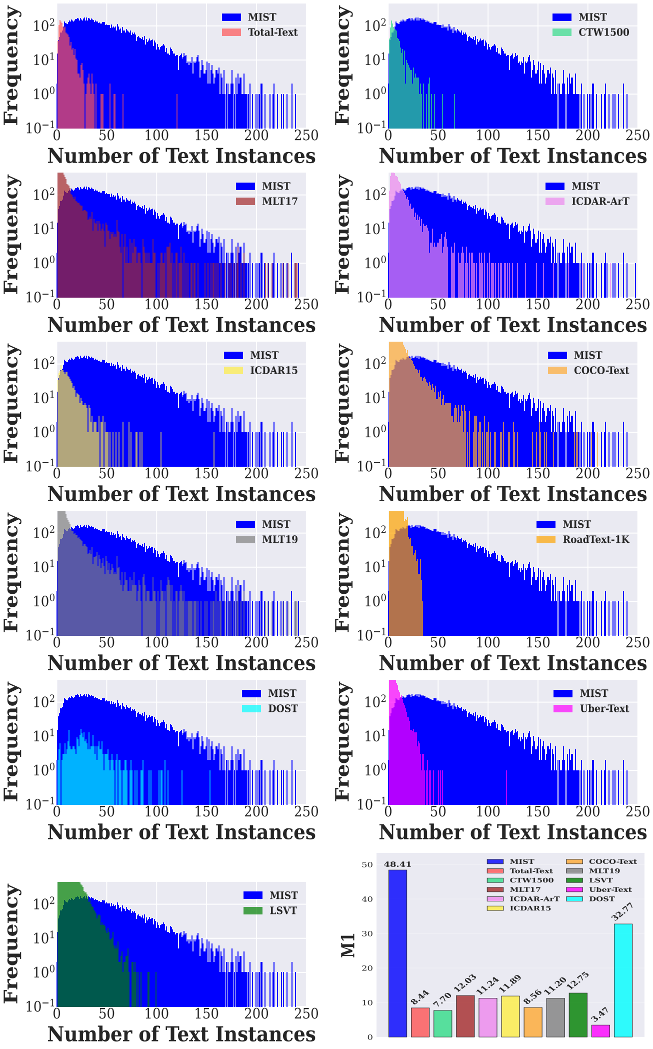


Figure 1. Compares the **distribution of text instances in scene images** (M_1) of MIST against existing datasets: Total-Text, CTW1500, MLT17, ICDAR-ArT, ICDAR15, COCO-Text, and RoadText-1K.

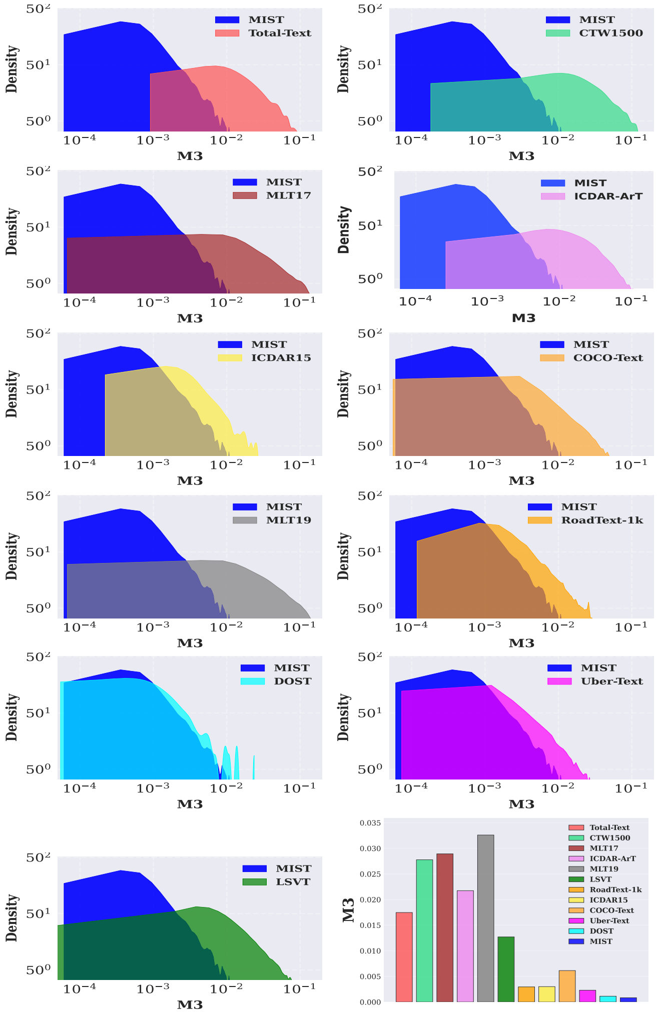


Figure 2. Compares the **average area of a text instance about the scene image** (M_3) of MIST against: Total-Text, CTW1500, MLT17, ICDAR-ArT, ICDAR15, COCO-Text, and RoadText-1K. The analysis employs kernel density estimation to create continuous distribution curves, with both axes displayed on logarithmic scales – base 50 for the y-axis and base 2 for the x-axis.

B. Dataset Comparison Complete

In terms of the occlusion/coverage metric M_3, DOST and Uber-Text approach MIST, but both have practical limitations. **DOST**, the closest to MIST in its incidental capture, offers only a static-image split with **338 training images**. Additionally, its frames are subsampled at every 10th interval, introducing redundancy and limiting diversity. **Uber-Text**, on the other hand, has a very low average of ≈ 3 text instances per image, so it is used more often for scene text *recognition* than for detection.

C. Evaluation Metrics

We adopt the DetEval protocol for evaluating scene text detection models, as used in ICDAR15 and Total-Text. This protocol employs three matching strategies — One-to-One, One-to-Many, and Many-to-One.

One-to-One: This matching strategy is used when a detected instance and a ground truth instance uniquely correspond to each other within the specified thresholds, with no additional overlapping candidates.

One-to-Many: Once all One-to-One matches have been identified, this matching algorithm assigns a single ground truth instance to multiple detections. This approach ensures that multiple valid segmented detections related to a single ground truth are not penalized, which is especially important for long or multilingual text instances. Each segmented detection and its corresponding ground truth receive a partial score. This scoring acknowledges the partial correctness of the detection and the degree of information captured.

Many-to-One: After completing the previous matches, this algorithm assigns a single detection to multiple ground truths. This issue commonly arises in crowded environments, where several text instances may be inaccurately identified as just one detection. Similar to one-to-many matching, both the ground truths and the single detection receive partial scores.

D. Training Configurations

The training code for the compared methods can be found at the following repositories: DBNet++[1], MixNet[2], TextBPN++[3], and DPText-DETR[4].

In Table 1, the training configurations for the TextBPN++ models specific to other datasets have been provided. We used these models in Sec 3.5 and Sec 5.2 in the main paper.

Fine-tuning dataset	Pretraining used
Total-Text [1]	MLT17 [4]
CTW1500 [3]	MLT17 [4]
17 [4]	SynthText [2]

Table 1. Training configurations for TBPN models for Total-text, CTW1500 and MLT17

[1] https://github.com/MhLiao/DB
[2] https://github.com/D641593/MixNet
[3] https://github.com/GXYM/TextBPN-Plus-Plus/
[4] https://github.com/ymy-k/DPText-DETR

Learning rate and schedule for TextBPN++ and DPText-DETR : We optimize the network with an initial learning rate $\eta_0 = 1 \times 10^{-4}$. A step–decay scheduler is used: at fixed intervals of s epochs, the learning rate is multiplied by a decay factor $\gamma = 0.9$, i.e.,

$$\eta_t \; = \; \eta_0 \cdot \gamma^{\lfloor t/s \rfloor},$$

where t counts epochs and s is the step size.

Hyperparameter	**Value**
Initial learning rate	1×10^{-4}
Learning rate scheduler	Step decay
Decay factor (per step)	0.9

Table 2. Optimization hyperparameters.

E. Insights and Takeaways

Generalization Capability of MIST

- We manually re-annotated CTW1500 at the word level for a fair comparison, as its official annotations are at the text-line/phrase level. We exclude CTW1500 and ICDAR-ArT from training to avoid train–test leakage arising from their line/phrase-level annotations and overlaps with other benchmarks.
- Because MLT17 is a competition dataset with a non-public test set, we use the provided validation set to analyze metrics and fine-tuning performance.
- We do not report the performance of a model trained on ICDAR-ArT when evaluated on Total-Text or CTW1500, since the ICDAR-ArT training set contains samples overlapping the test sets of those datasets, which would confound evaluation.

Transfer Learning Capability Score

Table 3 summarizes the performance of DPText-DETR and TextBPN++ pre-trained on MIST and fine-tuned on Total-Text, CTW1500, and MLT17.

Few Shot Transfer Learning Capability

We demonstrate the few-shot transfer learning capabilities of TextBPN++ and DPText-DETR, pre-trained on MIST, when applied to existing datasets such as CTW1500 and Total-Text. The training configurations remain the same as in the main paper.

CTW1500: We conduct transfer learning on TextBPN++ using samples of 100, 500, and 1000 from CTW1500. As shown in Figure 3(a), the model consistently surpasses the baseline performance of TextBPN++ on CTW1500, as reported in the original paper. Notably, even with just 100 samples, the model performs exceptionally well. However, its best performance is achieved with the largest sample size, surpassing the base performance by 1.20%

Test Set	**Model**	**P**	**R**	**F**	**F$^\alpha$**
Total-Text	H1	92.06	87.05	**89.48**	89.00
	H2	92.60	88.85	**90.69**	90.13
CTW1500	H1	91.58	86.72	**89.08**	88.80
	H2	88.56	86.83	**87.69**	86.49
MLT	H2	91.57	74.29	**82.02**	81.19

Table 3. Shows performance of DPText-DETR [5] and TextBPN++ [6] pre-trained on MIST and fine-tuned on Total-Text and CTW1500. H1 and H2 indicate DPText-DETR and TextBPN++, respectively. F$^\alpha$ is reported F-Measure for Total-Text, CTW1500, and MLT17 using these models without pre-training on MIST. Bold and underscore indicate the best and second-best value, respectively. Since DPText-DETR is not evaluated on MLT17, we only use TextBPN++ for MLT17.

Sample Size	**F-Measure (Total-Text)**	**F-Measure (CTW1500)**
100	85.10	82.24
500	86.90	86.43
1000	87.63	87.47
2000	88.98	88.06
4000	89.04	88.67
8000	89.20	89.02
Full	**89.48**	**89.08**

Table 4. F-Measure across different sample sizes for two evaluation settings. The final row indicates performance when using the entire dataset (17,600 and 13,000 samples respectively).

We perform transfer learning on DPText-DETR using sample sizes of 100, 500, 1000, 2000, 4000, 8000, and 13,000 as shown in Figure 3(b). The samples were sourced from data provided by the authors of TextBPN++ and DPText-DETR, resulting in different sample sizes. The model surpasses DPText-DETR's best-reported performance on CTW1500 by 0.28%, a notable achievement given that DPText-DETR is the state-of-the-art model for this dataset. Remarkably, it even surpasses the baseline performance with just 8000 training samples and achieves near baseline performance with 4000 samples. This further demonstrates the ability of MIST trained models to adapt to specific domains.

Total-Text: For Total-Text, we conduct transfer learning on TextBPN++ using sample sizes of 125, 625, and 1255. Our model surpasses the base performance of TextBPN++, as reported in its original paper (see Figure 4 (a)). MIST achieves an F-measure close to the base performance even

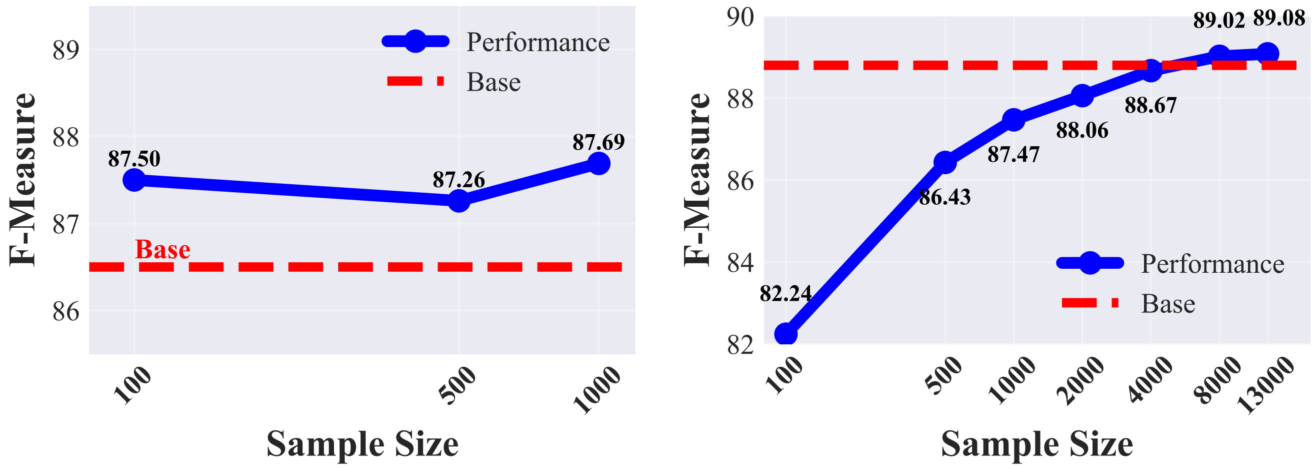


Figure 3. (a) shows the few-shot transfer learning performance of TextBPN++ pre-trained on MIST for CTW500. (b) illustrates the few-shot transfer learning performance of DPText-DETR pre-trained on MIST for CTW1500.

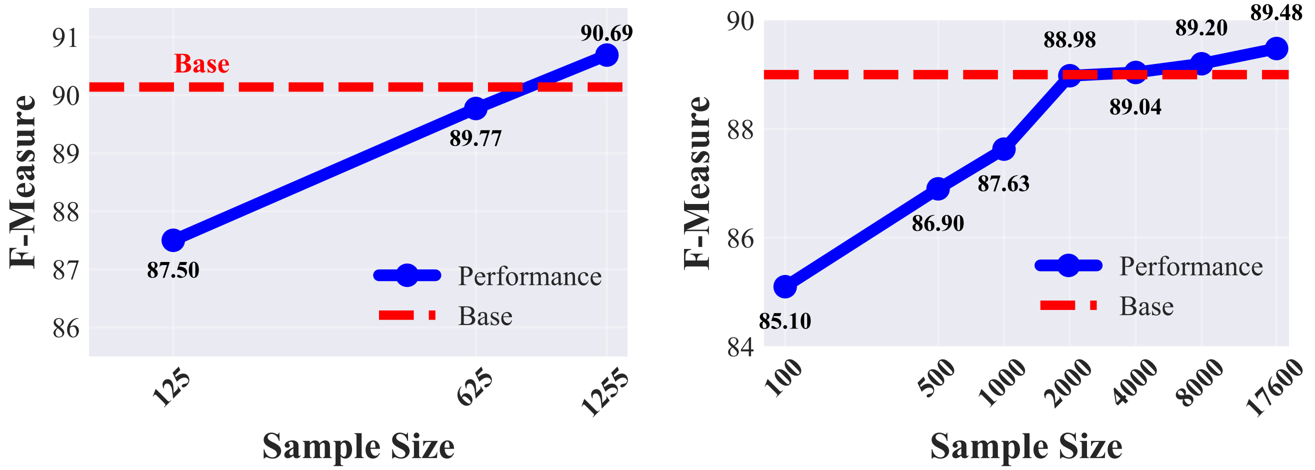


Figure 4. (a) shows the few-shot transfer learning performance of of TextBPN++ pre-trained on MIST for Total-Text. (b) illustrates the few-shot transfer learning performance of DPText-DETR pre-trained on MIST for Total-Text.

with a small sample size, demonstrating its strong adaptability to specific domains.

For DPText-DETR, transfer learning is performed with sample sizes of 100, 500, 1000, 2000, 4000, 8000, and 17,600. The model reaches performance close to its baseline with just 2,000 out of the 17600 samples, falling short by only 0.02%. Beyond this point, increasing the amount of training samples to 4000 and 8000 surpasses the base performance. This reinforces the adaptability of MIST-trained models on different data distributions (see Figure 4 (b)).

In addition to its value as an incidental scene text dataset and benchmark, MIST stands out as the most generalizable dataset in the literature, well-suited for real-world challenges and highly effective for pretraining. Its strong pretraining performance stems from the diverse and complex text instances it captures, providing the model with samples to tackle macro-level disturbances. Additionally, MIST can achieve near-baseline performance on specific datasets with minimal training data, highlighting its exceptional value in scene text research.

Issue Mitigation

In the main paper, we raised the topic of issue mitigation by fine-tuning a MIST pre-trained model. This section explores issue mitigation on MLT17 and the metric M_2. Through Figure 5, the effect of the MIST-pre-trained model was seen, improving TextBPN++'s performance on relatively lower M_3. We perform the same experiment

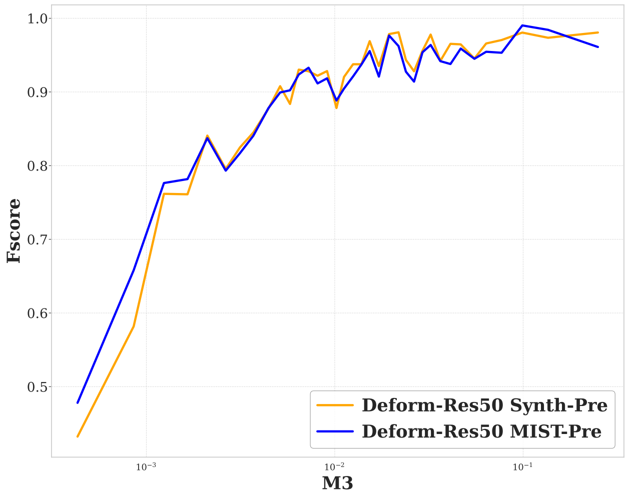


Figure 5. Compares the F-measure and M_3 of TextBPN++ pre-trained with SynthText and MIST, trained on MLT17, while evaluating on MLT17.

on MLT17, grouping the MLT17's samples into batches of 50 and calculating the F-measure for each batch. In the case of MLT17, the mitigation of the issue is ambiguous. The MIST pre-trained model initiates with improved performance on lower M_3; however, after the first 4 batches, the performance becomes ambiguous, with relatively fewer batches corresponding to improved performances than Total-Text. Out of the first 10 batches, MIST pre-trained models outperform the base SynthText pre-trained model 6 times. When coupled with the model outperforming the base TextBPN++ model on the validation set, this observation suggests a positive result of issue mitigation at smaller M_3.

We perform a similar analysis on M_2 by labeling text instances larger than size thresholds as 'do not care' regions and calculating the F-measure for the filtered test set. We plot the F-Measure against M_2, as seen in Figure 6. Unlike the performance on M_3, where MLT17 was ambiguous, here, the performance on MLT17 surpasses the base model's performance on all thresholds. In contrast, the model fine-tuned on Total-Text is extremely ambiguous at all thresholds.

MIST's issue mitigation abilities at various scales are highlighted through these observations. This property can be attributed to its collection of varied text instances and many scenes in harsh conditions.

F. VLM performance on MIST

We qualitatively evaluated general-purpose vision language models (VLMs) ChatGPT-5 and Gemini 2.5 Pro on 5 MIST test images to probe their ability to perform scene text detection directly from prompts. While these VLMs often demonstrate strong high-level image understanding, their detection outputs on MIST were underwhelming and exhibited several recurrent issues:

- **Sparse and low-recall detections.** Models produced a significantly low recall rate of 15%, missing the majority of text instances.
- **Resolution/scale mismatch.** The coordinates returned by the models appeared to be defined in the (downsampled/compressed) internal inference resolution rather than the original image size; without explicit post-processing. This led to systematic misalignment of boxes when overlaid on the 1920×1080 input.

Visual results provided in Fig. 7

G. Visual Samples and Results

A Few Samples from MIST

Figure 8 shows a few sample images from MIST with diverse text instances.

Comparison of Visual Result

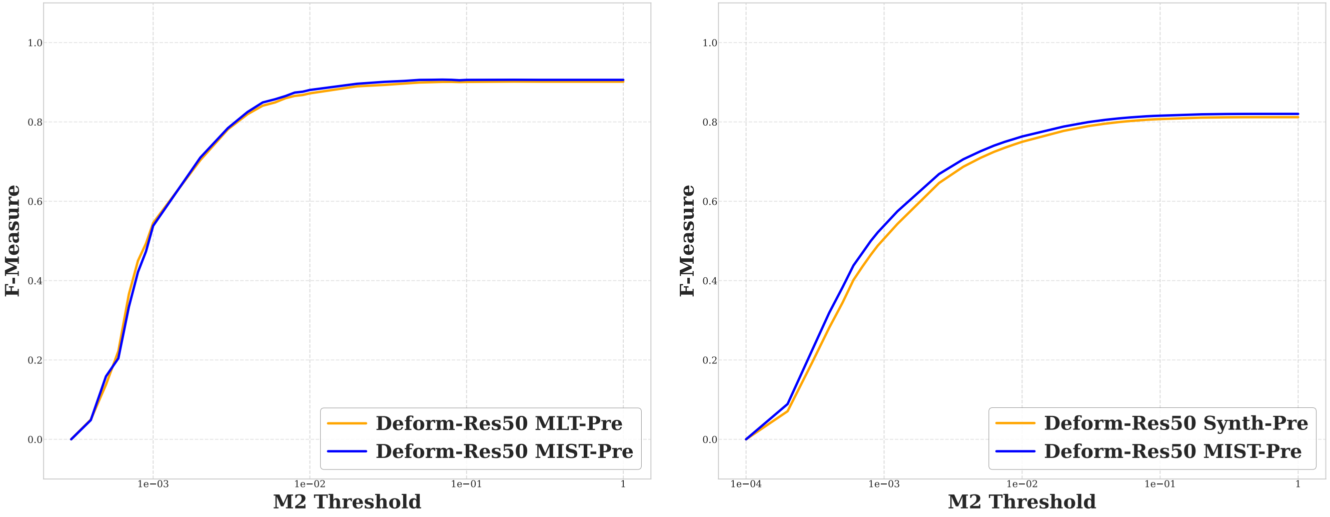
Figure. 9 visually compares text detection results in MIST by (a) TextBPN++ trained on MIST, (b) state-of-the-art model for Total-Text, (c) state-of-the-art model for ICDAR-ArT, (d) state-of-the-art model for MLT.

Visual Result of Cross-Domain Scene Images

Figure 10 shows detected text in Total-Text and MLT by TextBPN++ trained on MIST, state-of-the-art model for Total-Text, and state-of-the-art model for MLT.

Visual Results of Out-of-Domain Scene Images

We download a few scene text images from the Internet by searching. We provide text detection results of TextBPN++ trained on MIST, Total-Text, MLT, and ICDAR-ArT. Figure 11 shows detected text using different models. It is visually shown that TextBPN++ trained on MIST is able to detect more text than other models.



(a) (b)

Figure 6. (a) Compares the F-measure and M_2 of TextBPN++ pre- trained with MLT17 and MIST, trained on Total-Text, while evaluating on Total-Text. (b) Compares the F-measure and M_2 of TextBPN++ pre-trained with SynthText and MIST, trained on MLT17, while evaluating on MLT17.

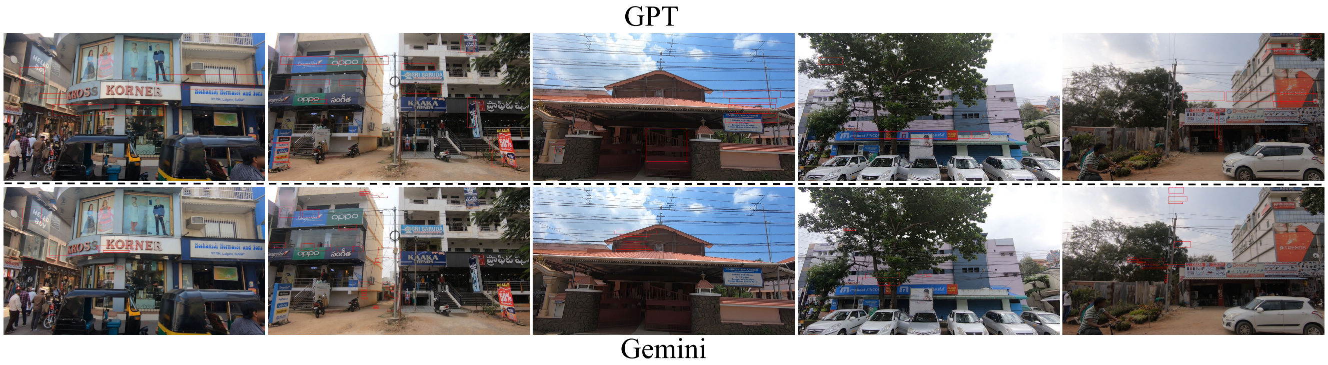


Figure 7. The predicted boxes are annotated in red. As we can see, the boxes are not aligned with the text. However, there seems to be some sign that the models understand the layout of the text and were constrained due to post-processing.

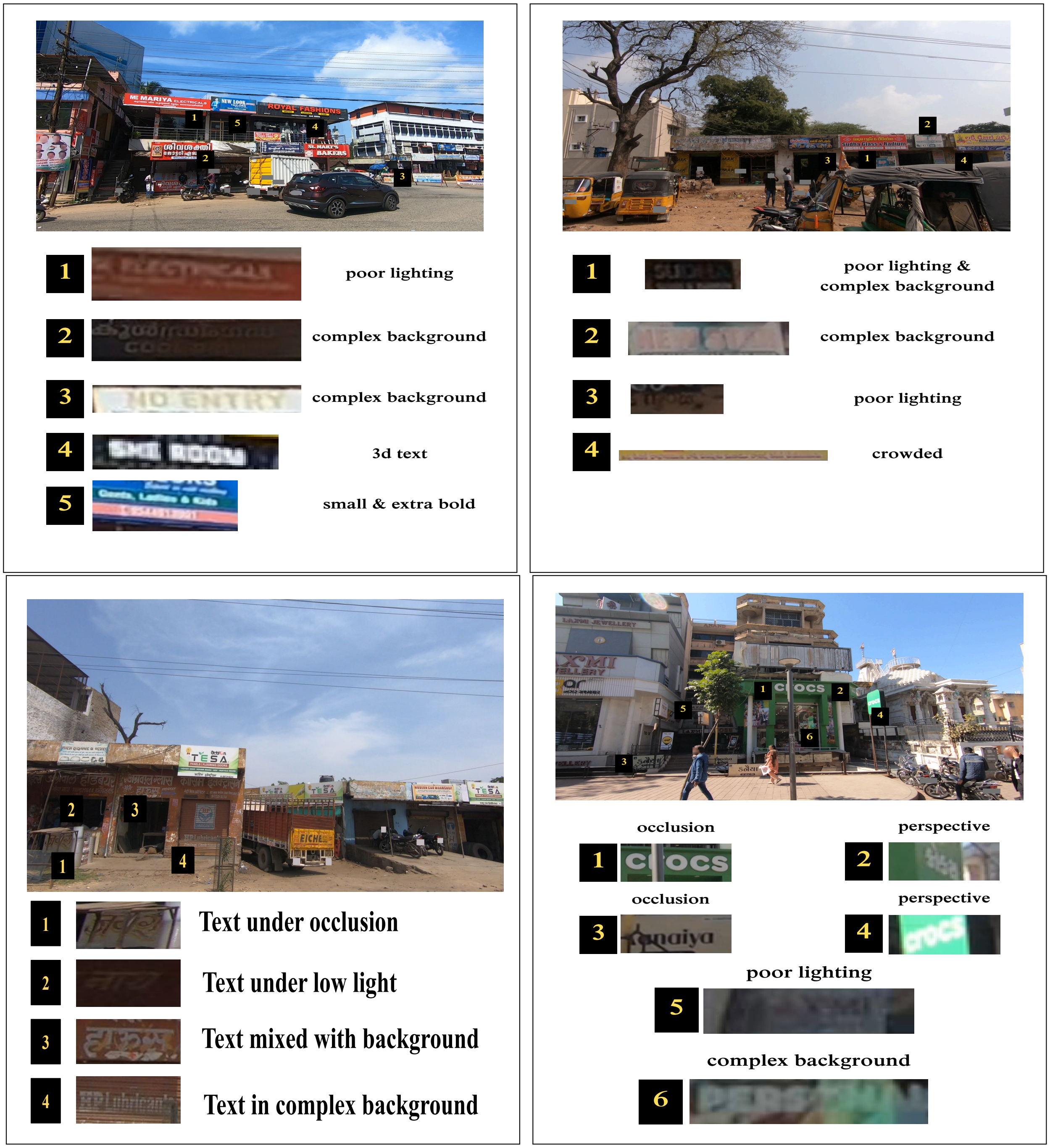
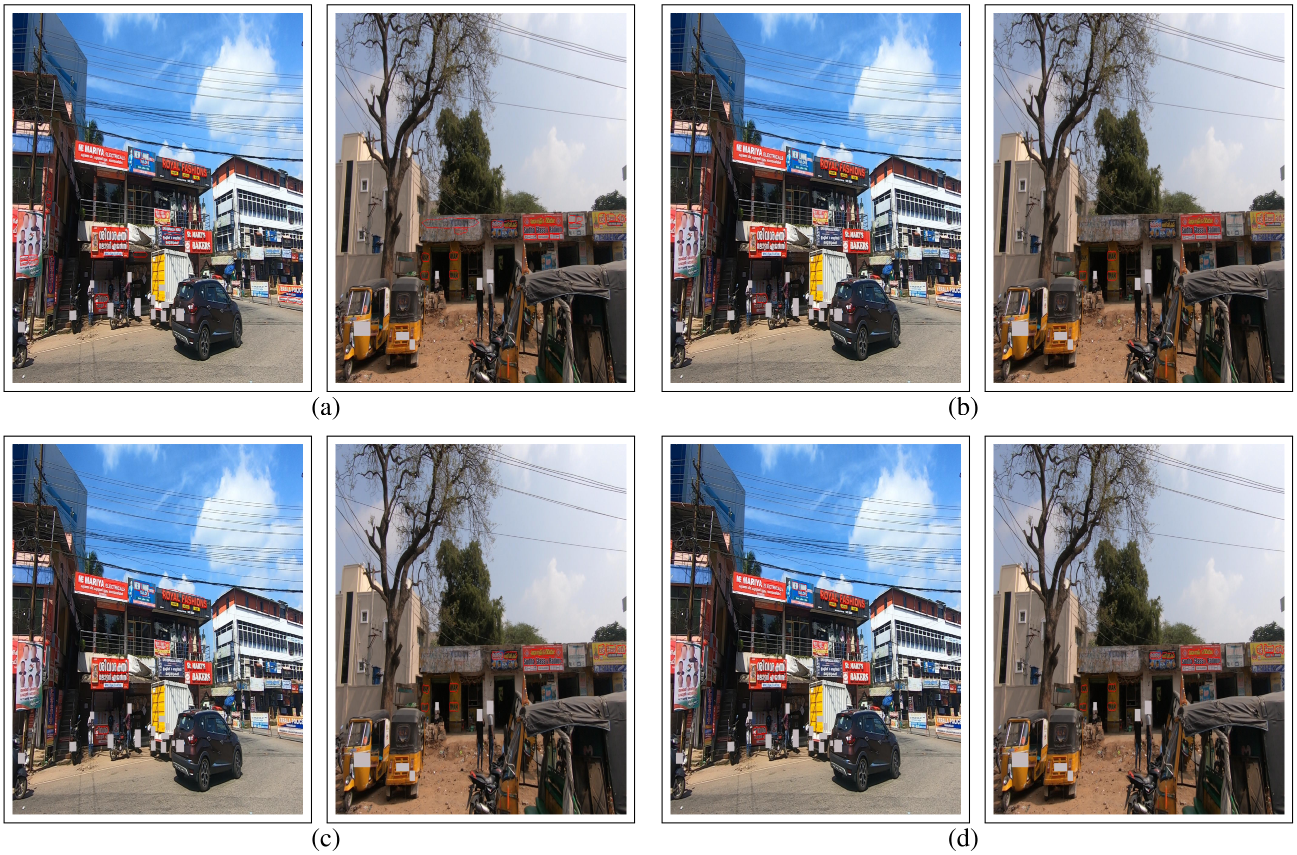
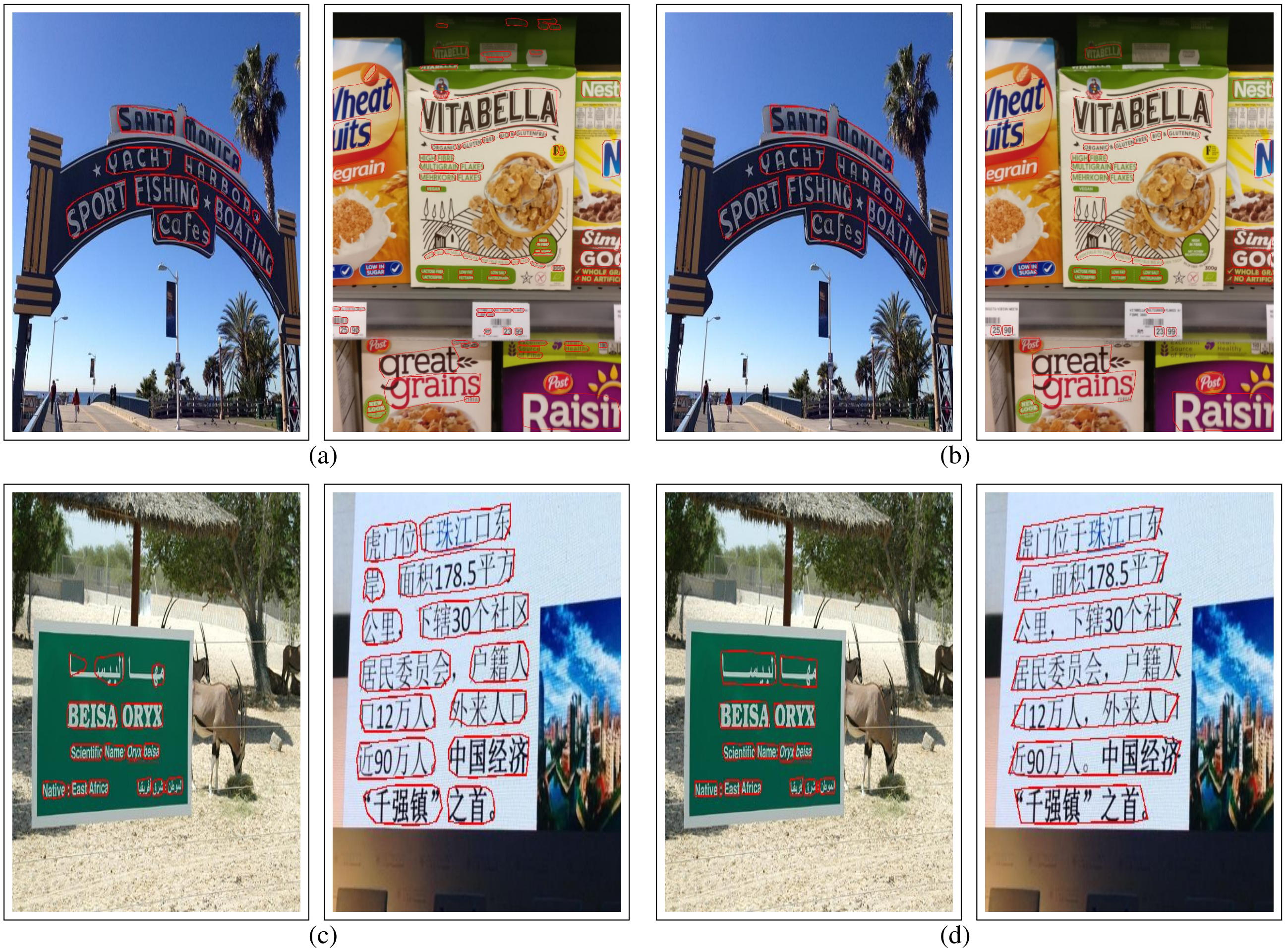


Figure 8. Shows sample images containing text instances in poor light, complex background, crowded surrounding, 3D and smaller text, occlusion, perspective distortion.



(a) (b) (c) (d)

Figure 9. Shows predicted text detection results of MIST. (a) detected text by TextBPN++ trained on MIST. (b) detected text by state-of-the-art model for Total-Text. (c) detected text by state-of-the-art model for ICDAR-ArT. (d) detected text by state-of-the-art model for MLT.



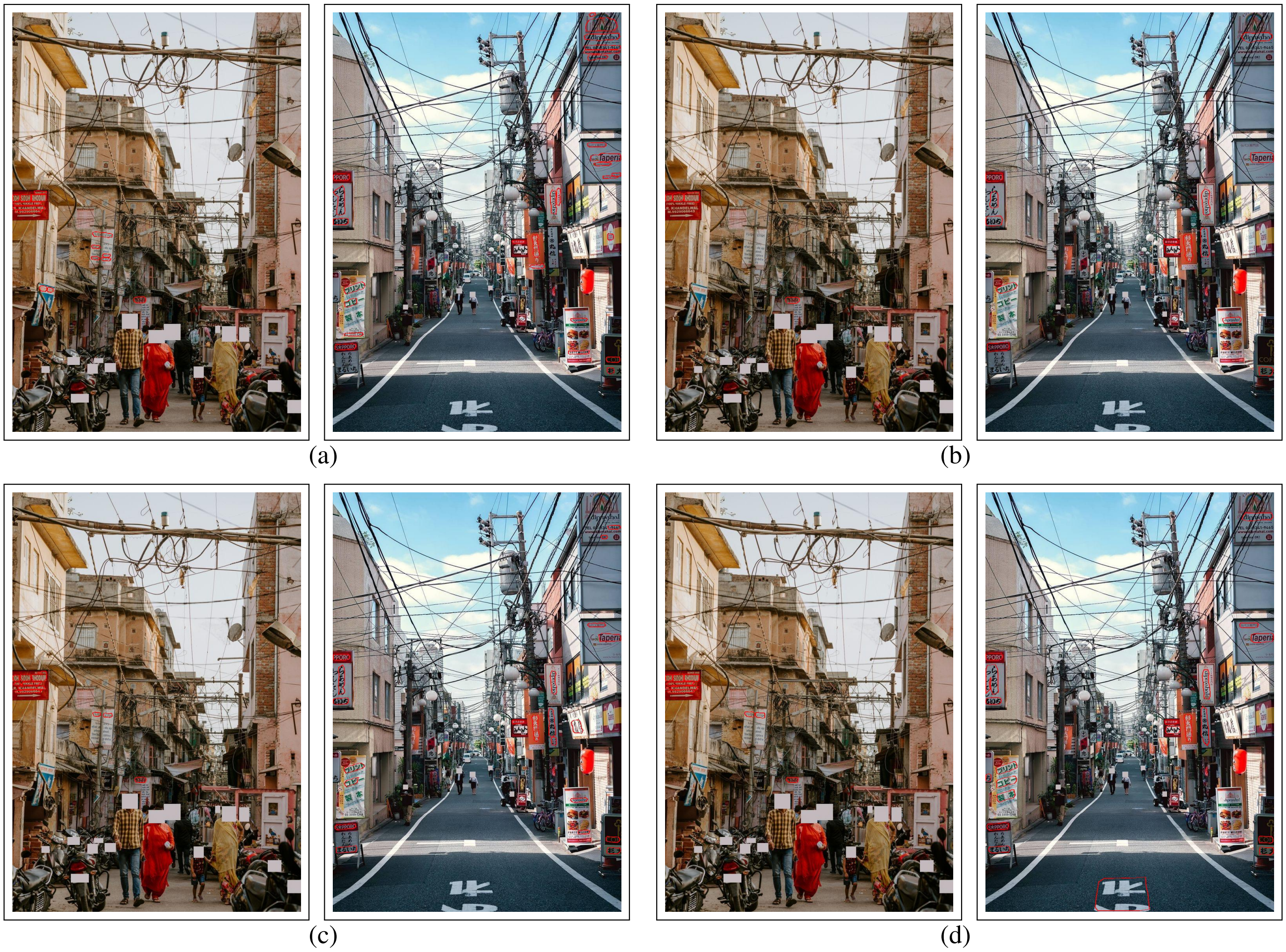
(a)

(b)

(c)

(d)

Figure 10. Shows predicted text detection results of Total-Text and MLT. (a) detected text in Total-Text by TextBPN++ trained on MIST. (b) detected text in Total-Text by state-of-the-art model for Total-Text. (c) detected text in MLT by TextBPN++ trained on MIST. (d) detected text in MLT by state-of-the-art model for MLT.



(a)

(b)

(c)

(d)

Figure 11. Shows predicted text detection results on out-of-distribution scene images. (a) detected text by TextBPN++ trained on MIST. (b) detected text by state-of-the-art model for Total-Text. (c) detected text by state-of-the-art model for MLT. (d) detected text by state-of-the-art model for ICDAR-ArT.

References

[1] Chee Kheng Ch'ng and Chee Seng Chan. Total-Text: A comprehensive dataset for scene text detection and recognition. In *ICDAR*, pages 935–942, 2017.

[2] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *CVPR*, pages 2315–2324, 2016.

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

[4] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, et al. ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification-RRC-MLT. In *ICDAR*, pages 1454–1459, 2017.

[5] Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Bo Du, and Dacheng Tao. DPText-DETR: Towards better scene text detection with dynamic points in transformer. In *AAAI*, pages 3241–3249, 2023.

[6] Shi-Xue Zhang, Chun Yang, Xiaobin Zhu, and Xu-Cheng Yin. Arbitrary shape text detection via boundary transformer. *TMM*, 26:1747–1760, 2023.