Tree-based Manga Layout Generation: A Comparative Analysis of Visual and Textual Features

Anonymous ACL submission

Abstract

Panel layout plays a crucial role in how manga pages convey narrative structure, influencing pacing, emphasis, and reading flow. Despite its importance, page-level layout has rarely been the primary modeling target in computational research. This paper investigates whether layout structure can be inferred from panel content representations and how it supports geometric layout generation. We compare visual features extracted from panel images with textual descriptions generated by a large multimodal model, using a unified framework to predict hierarchical layout. Our analysis reveals a consistent modality gap: visual representations enable more reliable layout inference, while textual descriptions provide weak structural cues. Based on these findings, we propose a two-stage framework: first predicting a layout tree and then generating panel bounding boxes conditioned on the predicted structure. This structure-conditioned generation improves geometric accuracy and degrades gracefully when using predicted rather than ground-truth trees. We also introduce Manga109Caption, a new dataset extending Manga109 with panel-level captions for 109 titles.Our programs and datasets are available from [anonymous link].

1 Introduction

Manga are a multimodal medium where images, text, and spatial composition shape narrative meaning. A manga page is a organized sequence that controls pacing, emphasis, and reader attention. Previous studies (McCloud and Martin, 1993; Cohn, 2013) have shown that the *layout* of panels—how they are spatially organized—plays a crucial role in shaping narrative experience, influencing temporal progression, causality, and emphasis.

Despite its importance, panel layout has received limited attention in computational research. Most existing work on comic analysis focuses on local tasks, such as speech balloon detection (Dubray and Laubrock, 2019), character identification (Sachdeva and Zisserman, 2024), or speaker identification (Li et al., 2024), treating page-level layout as a given rather than a modeling target. Feng et al. (2025) shows that manga can often be identified from its layout alone, indicating that panel layout encodes work-specific characteristics. Recent advances in manga generation have concentrated on panel-level image synthesis (Wu et al., 2025). Even systems generating full pages (Chen et al., 2024) treat layout generation implicitly, making it difficult to analyze how page-level structure is determined. Most generation pipelines rely on textual descriptions that focus on narrative content—describing what happens within panels or across a page. This raises the question: do such descriptions provide sufficient cues to determine *how* panels should be spatially organized—that is, whether page-level layout can be recovered from panel content descriptions alone?

Recent advances in large multimodal models (LMMs), such as LLaVA (Liu et al., 2023), GPT-4V (Achiam et al., 2023), and Qwen-VL (Yang et al., 2025), have significantly improved image-to-text generation. These models produce detailed captions for individual images. However, it remains unclear whether such descriptions capture the structural information necessary to organize multiple panels into a coherent layout. Manga provide an ideal testbed for this issue, as their meaning arises not only from panel content but also from spatial relationships. If textual descriptions could encode this structure, page organizations should be recoverable from the text. We investigate this by comparing how visual and textual modalities support the inference of hierarchical organization in manga page layouts.

Our experiments reveal a clear difference between visual and textual modalities in their ability to support layout generation. Neither visual features nor textual descriptions alone can directly

reproduce the spatial coordinates of a manga page. However, when using visual features extracted from panel images, even relatively lightweight models can reliably infer an intermediate structural organization of the page, which in turn enables the generation of geometrically coherent and accurate layouts. In contrast, textual descriptions that primarily focus on panel content—even when produced by powerful large multimodal models—appear insufficient to reliably determine how panels should be arranged within a page, which makes direct text-to-layout generation challenging in this setting.

Motivated by these observations, we propose a two-stage framework for *hierarchical manga layout generation*. In the first stage, the model predicts an intermediate structural representation from visual features, capturing how panels are hierarchically organized on the page. In the second stage, this structure guides the realization of concrete page geometry, producing coherent panel arrangements that respect the inferred organization. To facilitate further research on the relationship between narrative modality and spatial composition, we also introduce a new dataset derived from the Manga109 corpus (Matsui et al., 2017; Aizawa et al., 2020), which provides descriptive captions for every panel and enables controlled comparison between visual and textual modalities.

Our contributions are threefold:

- We empirically demonstrate that visual features are more effective than content-oriented textual descriptions in supporting the inference of manga page structure, highlighting differences in how visual and linguistic modalities encode narrative organization.
- We propose a structure-guided framework that generates manga page layouts from sequences of panel images through an explicit intermediate structural representation.
- We present **Manga109Caption**, a dataset based on the Manga109 corpus with panel-level descriptive captions to support future research on multimodal narrative and layout understanding.

2 Related Work

2.1 Comics Computing

Research on computational approaches to comics has expanded rapidly in recent years. Much of this work has addressed low-level perception tasks such as speech balloon detection (Rigaud et al., 2021; Dubray and Laubrock, 2019) and optical character recognition (Baek et al., 2022), which provide the basic elements for higher-level analysis. At the narrative level, studies have investigated key entities in comics, including character recognition across panels (Li et al., 2021) and speaker attribution for dialogue balloons (Sachdeva and Zisserman, 2024; Li et al., 2024). These tasks link visual and textual elements to narrative entities, forming the basis for modeling character interactions and story progression. While such directions have advanced the analysis of existing comics, they remain primarily concerned with decomposition and understanding rather than the generation of new content. In particular, layout generation—the mechanism that controls pacing and visual flow—has received little attention and remains largely unexplored.

2.2 Manga Generation

Recent advances in manga generation have achieved impressive results in synthesizing high-quality illustrations (Wu et al., 2025; Chen et al., 2024). However, page-level layout is often not explicitly modeled.

For example, Wu et al. (2025) demonstrates visually appealing manga generation and releases the MangaZero dataset with panel-level captions. However, their method requires users to manually specify panel number, sizes, and positions, leaving layout design outside the scope of the model.

Similarly, Chen et al. (2024) introduce the Manga109Story dataset and propose an end-to-end story-to-manga generation framework based on diffusion models. While their system generates full manga pages, it does not explicitly represent layout, making it difficult to analyze how layout decisions are made. Additionally, the Manga109Story dataset is not publicly available.

Taken together, these works highlight that, despite substantial progress in manga generation, page-level layout has not been explicitly treated as a primary generation target. In this work, we focus on single-page manga layout generation as an explicit objective, bringing panel arrangement into direct modeling and evaluation.

2.3 Layout Generation in Other Domains

Layout generation has been extensively studied in domains such as user interfaces, documents, and posters. For example, the RICO

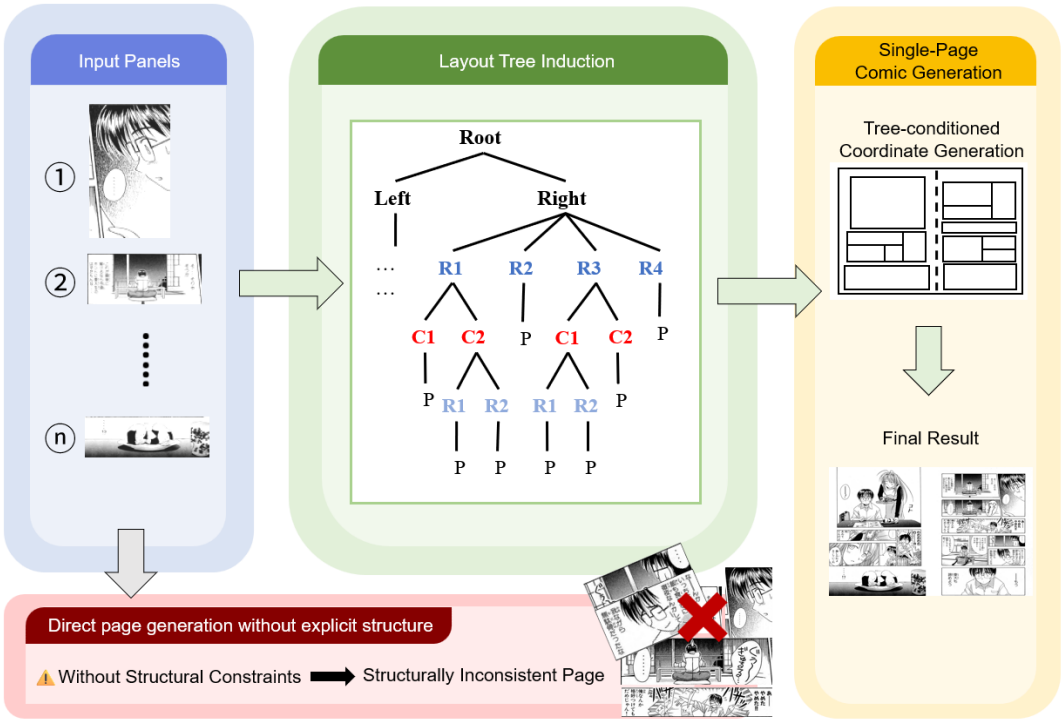

Figure 1: Overview of the proposed two-stage manga page layout generation framework. Given an ordered sequence of panels, the model first predicts a hierarchical layout tree that captures the global page structure, and then generates panel bounding boxes conditioned on the predicted tree. (Love Hina © Ken Akamatsu)

dataset enables UI layout modeling from structured widget specifications (Deka et al., 2017), and PubLayNet and DocBank provide large-scale document layout annotations for template-based or learned mappings (Zhong et al., 2019; Li et al., 2020). In text-to-layout, models like LayoutTransformer (Gupta et al., 2021), Text2Layout (Takahashi and Kuriyama, 2024), and two-stage pipelines that parse text before placement (Liang et al., 2023; Lin et al., 2023) have shown strong results on COCO, RICO, and Web5K. More recently, LLM-based methods such as LayoutGPT (Feng et al., 2023), PosterLLaVA (Yang et al., 2024), LayoutCoT (Shi et al., 2025), and PosterO (Hsu and Peng, 2025) have achieved state-of-the-art results on poster and advertising benchmarks by generating content-aware layouts through multimodal reasoning and exemplar retrieval. While these works demonstrate effective layout modeling in constraint-rich domains, they rely on explicit structural cues, which are often provided in document and UI generation. In manga, however, layout structure emerges implicitly from narrative progression and visual composition, making the inference problem qualitatively different.

3 Method

3.1 Overview

We formulate manga page layout generation as a two-stage problem: **(i) structural induction** and **(ii) geometric realization**. Given an ordered sequence of N panels, we first predict a *layout tree* that represents the page's recursive partitioning, and then generate bounding boxes conditioned on the predicted tree. An overview of this process is shown in Figure 1.

This separation enables the model to (a) capture global page organization and (b) generate stable panel coordinates, while allowing flexibility for local refinement.

Let $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ denote panel-level input features, where each $\mathbf{x}_i$ is either a visual embedding extracted from the i-th panel image or a textual embedding derived from its caption. The model outputs (1) a layout tree $\mathcal{T}$ for the full page and (2) a set of panel bounding boxes $\mathbf{B} = \{b_i\}_{i=1}^{N}$, where $b_i = (x_{\min}, y_{\min}, x_{\max}, y_{\max})$.

3.2 Layout Tree Representation

Following prior work (Cao et al., 2012) on manga layout modeling, we represent a page layout as a hierarchical structure induced by recursive spatial

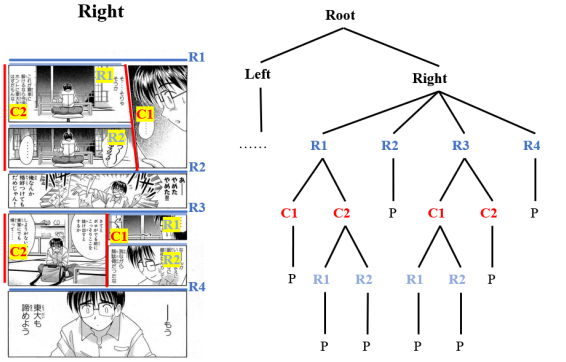

Figure 2: An example of recursive spatial subdivision of a manga page (left) and the corresponding layout tree representation (right). In this figure, only the right half of the manga page is shown. The page is first divided into left and right regions, and is then recursively partitioned using vertical splits (R) and horizontal splits (C), where each leaf node (P) corresponds to an individual panel. (Love Hina © Ken Akamatsu)

subdivision. As shown in Figure 2, the root node ROOT splits the page into two major regions, RIGHT and LEFT, reflecting the organization of manga pages. Within each region, the layout is recursively partitioned by two types of internal nodes: R (row split) and C (column split). Leaf nodes correspond to panels, denoted by the token P.

To enable sequence modeling, we linearize the layout tree as a bracketed pre-order traversal, i.e., an S-expression over the vocabulary $\mathcal{V} = \{\texttt{ROOT}, \texttt{RIGHT}, \texttt{LEFT}, \texttt{R}, \texttt{C}, \texttt{P}, (,)\}$. The resulting token sequence is denoted as $\mathbf{Y} = (y_1, \ldots, y_T)$. Under our layout tree definition, leaf nodes are ordered according to the panel reading order. As a result, the k-th P token in $\mathbf{Y}$ corresponds to the k-th panel in $\mathbf{X}$, without requiring any additional alignment. This enables seamless transfer of structural information to the coordinate generation stage.

3.3 Sequence-to-Sequence Tree Generation

Problem formulation. We predict a layout tree token sequence $\mathbf{Y}$ conditioned on the ordered panel sequence $\mathbf{X}$:

$$p(\mathbf{Y} \mid \mathbf{X}) = \prod_{t=1}^{T} p(y_t \mid y_{<t}, \mathbf{X}). \tag{1}$$

Encoder. We encode the panel sequence using a bidirectional recurrent encoder. Each input feature $\mathbf{x}_i$ is projected into a shared hidden space, and a BiLSTM produces contextualized representations $\mathbf{H} = (\mathbf{h}_1, \ldots, \mathbf{h}_N)$, which summarize both local and global panel context.

Decoder. An autoregressive Transformer decoder generates the tree token sequence by attending to the encoder representations $\mathbf{H}$ at each step. The model is trained using standard cross-entropy loss:

$$\mathcal{L}_{\text{tree}} = -\sum_{t=1}^{T} \log p(y_t^* \mid y_{<t}^*, \mathbf{X}), \tag{2}$$

where $\mathbf{Y}^*$ denotes the ground-truth tree sequence.

Grammar-constrained decoding. To ensure that predicted sequences correspond to valid layout trees, we apply grammar-constrained decoding at inference time. Specifically, we enforce: (i) well-formed bracket structure (balanced parentheses), and (ii) subtree-wise leaf-count consistency, such that the number of P tokens assigned to each page half (LEFT and RIGHT) matches the corresponding number of input panels. These constraints are implemented as a dynamic hard mask over the decoder's next-token distribution, restricting generation to a finite-state grammar. As a result, every decoded sequence can be deterministically parsed into a valid layout tree.

3.4 Tree-Conditioned Coordinate Generation

Model. Given a predicted layout tree $\mathcal{T}$, we generate panel bounding boxes in a structure-guided manner, followed by content-based refinement.

For each leaf panel, we derive a structural representation that encodes its position in the recursive subdivision hierarchy, including its split path and relative structural attributes. These structural representations are combined and processed by a Transformer encoder to model cross-panel dependencies and produce a set of base bounding boxes:

$$\hat{\mathbf{B}}^{\text{base}} = f_\theta^{\text{struct}}(\mathcal{T}), \tag{3}$$

where $\hat{\mathbf{B}}^{\text{base}} = \{\hat{b}_i^{\text{base}}\}_{i=1}^{N}$ captures the global page structure implied by the layout tree.

To account for local geometric variation, we further apply a content-based refinement module. Given panel content features $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, a second Transformer encoder predicts panel-wise adjustments:

$$\Delta\hat{\mathbf{B}} = f_\theta^{\text{content}}(\mathbf{X}). \tag{4}$$

The final panel bounding boxes are obtained by combining the structure-based predictions with the content-based adjustments:

$$\hat{\mathbf{B}} = \hat{\mathbf{B}}^{\text{base}} + \Delta\hat{\mathbf{B}}. \tag{5}$$

This design allows the model to preserve global layout consistency imposed by the predicted structure, while flexibly adapting panel geometry to local content cues.

Training objective. We train the model using a regression loss based on intersection-over-union, augmented with auxiliary losses:

$$\mathcal{L}_{\text{coord}} = \lambda_{\text{iou}}\mathcal{L}_{\text{IoU}} + \lambda_{\text{l1}}\mathcal{L}_{\text{SmoothL1}} + \lambda_{\text{res}}\mathcal{L}_{\text{residual}}. \tag{6}$$

Using predicted vs. ground-truth trees. To isolate the effect of structural induction quality, we evaluate coordinate generation under two conditions: conditioning on ground-truth layout trees and conditioning on predicted trees. This comparison quantifies how errors in structure prediction propagate to geometric realization.

4 Dataset: Manga109Caption

4.1 Overview and Motivation

To investigate how visual and textual modalities differ in their ability to support manga layout generation, we require a dataset that aligns panel images with descriptive textual captions. Such paired data are scarce in existing public resources. While the **MangaZero** dataset (Wu et al., 2025) provides both panel bounding boxes and descriptive text, its bounding boxes are automatically detected using the Magi model (Sachdeva and Zisserman, 2024; Sachdeva et al., 2024; Sachdeva and Zisserman, 2025), and often fail to precisely delineate the true panel boundaries. In contrast, the **Manga109** dataset (Matsui et al., 2017; Aizawa et al., 2020) offers human-annotated panel bounding boxes, providing more reliable spatial information that better suits our analysis. Although the Manga109Story dataset (Chen et al., 2024) has enriched Manga109 with panel-level captions generated by large multimodal models, it is not publicly available. Therefore, to enable systematic study of how visual and textual inputs relate to manga layout structures, we construct a new dataset named **Manga109Caption**, which augments Manga109 with automatically generated descriptive captions for every panel.

4.2 Dataset Construction

To pair panel images with textual descriptions, we automatically generate narrative captions for the Manga109 dataset. Specifically, we use **LLaVA-v1.6-34B** (Liu et al., 2024) to describe the visual content of each panel image. For each panel, the model receives only the cropped panel image as input and outputs a descriptive sentence in English that focuses on depicted characters, actions, and settings, while deliberately excluding any explicit spatial or layout information.

We exclude four-panel (yonkoma) manga, as their fixed layout structure differs substantially from general manga pages.

In total, we generate captions for **109 manga titles**, covering **10,602 pages** and **103,849 panels**, resulting in a large-scale paired collection of panel images and textual descriptions. We split the dataset at the title level into training, development, and test sets with an 8:1:1 ratio.

5 Experiments

We evaluate the proposed two-stage framework on two tasks: (1) layout structure prediction and (2) layout coordinate generation.

5.1 Experimental Setup

All experiments are conducted on the Manga109Caption dataset introduced in Section 4. Unless otherwise specified, textual features are extracted using Sentence-BERT all-MiniLM-L6-v2 (Wang et al., 2020), and visual features are extracted using CLIP ViT-B/32 (Radford et al., 2021). Implementation details and hyperparameters are provided in the appendix.

5.2 Evaluation Metrics

Layout structure prediction is evaluated using **Tree Edit Distance (TED)** (Zhang and Shasha, 1989), which measures the minimum number of edit operations required to transform a predicted layout tree into the corresponding ground-truth tree. Lower TED indicates closer structural similarity.

Layout coordinate generation is evaluated using **mean Intersection over Union (mIoU)** (Everingham et al., 2010), computed between predicted and ground-truth panel bounding boxes. Higher mIoU indicates better geometric alignment.

5.3 Layout Structure Prediction

Task. Given an ordered sequence of panels on a page, the task is to predict a hierarchical layout tree that represents the recursive spatial subdivision of the page. We compare our sequence generation model against non-parametric baselines that retrieve an existing layout tree from the dataset, including random selection and k-nearest neighbor

Table 1: Main results for layout structure prediction.

Method	Input Modality	TED ↓
Random	–	0.40
kNN	Text	0.40
kNN	Visual	0.39
Ours	Text	0.33
Ours	Visual	**0.21**

Table 2: Analysis of input representations for layout structure prediction.

Category	Input Variant	TED ↓
Text	Weak prompt	0.33
Visual	CLIP ViT-L/14	0.20
Visual	MAE ViT-B/16	0.20
Visual	MAE ViT-L/16	0.18
Visual	High-frequency only	0.19
Visual	Low-frequency only	**0.15**

(kNN) retrieval based on either textual or visual feature similarity. For fairness, both baselines retrieve candidate trees only from pages that have the same number of panels in the left and right page regions as the target page, rather than from the entire dataset. These baselines assess whether layout structure can be recovered from feature similarity alone, without explicit structural modeling.

Main Results. We find that layout structure can be inferred more reliably from visual features than from textual descriptions, even when using the same model architecture. Table 1 reports the quantitative results. Text-based models achieve a limited improvement over non-parametric baselines, indicating that captions encode some weak structural signals. However, this improvement remains substantially smaller than that obtained using visual features, which lead to a much larger reduction in tree edit distance.

5.4 Analysis of Input Representations

To better understand what types of information support layout structure prediction, we analyze different input representations while keeping the model architecture fixed. In particular, we examine whether the observed performance gap between visual and textual inputs is attributable to limitations of the textual descriptions, or to the nature of layout structure itself.

Caption Degradation. To assess whether the limited performance of text-based models stems from the quality or style of the captions, we evaluate a degraded caption setting in which captions are generated using a weak prompt without summarization constraints. Results are reported in Table 2. Details of the degraded caption prompting strategy are provided in Appendix A.2.

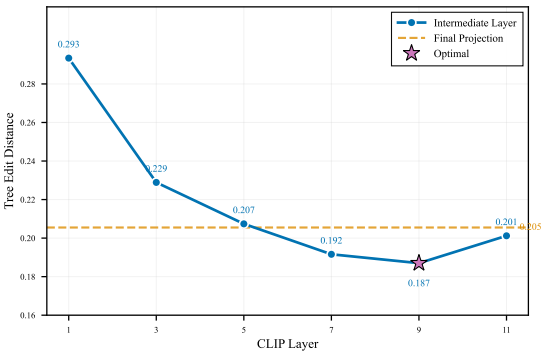

Figure 3: TED obtained using different layers of CLIP ViT-B/32.

(a) (b) (c)

Figure 4: Examples of frequency decomposition applied to a panel image: (a) original image, (b) low-frequency component, and (c) high-frequency component. (Love Hina © Ken Akamatsu)

Visual Encoder Variants. To examine how different visual pretraining objectives affect layout structure prediction, we compare alternative visual encoders, including CLIP ViT-L/14, which is trained with contrastive objectives, and MAE-based models (He et al., 2022) trained via masked reconstruction (Table 2). We further analyze representations extracted from different layers of CLIP ViT-B/32 (Figure 3).

Frequency Decomposition. To disentangle global geometric cues from local visual details, we apply 2D Fourier decomposition to panel images and evaluate models using only low-frequency or high-frequency components (Table 2). Examples of the resulting low- and high-frequency images are shown in Figure 4.

Overall, these analyses indicate that the observed

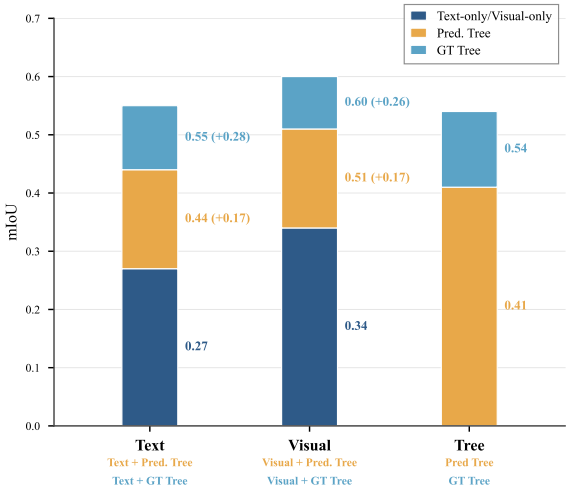

Figure 5: Coordinate generation results under three settings: without layout tree conditioning, with ground-truth layout trees (GT Tree), and with predicted layout trees (Pred. Tree).

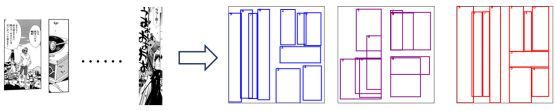

Figure 6: Qualitative comparison of panel coordinate generation results for the same page: ground-truth layout, prediction without layout tree conditioning, and prediction conditioned on a layout tree.

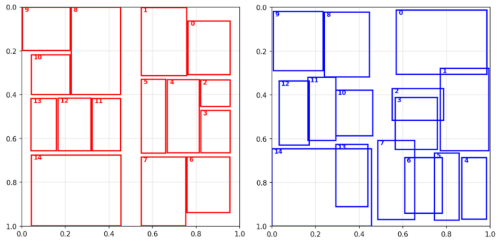

Figure 7: Comparison of coordinate generation results conditioned on ground-truth layout trees and on predicted layout trees, for the same input page.

performance gap between visual and textual inputs cannot be explained solely by caption quality. Performance varies substantially across visual representations—including different encoders, different CLIP layers, and low- vs. high-frequency inputs—whereas the degraded-caption setting does not materially change the text-based result.

5.5 Layout Coordinate Generation

Having established that layout trees can be reliably predicted, we next evaluate panel coordinate generation conditioned on the inferred layout structure.

Quantitative Results

Figure 5 reports mIoU under different conditioning settings. Conditioning on ground-truth layout trees yields a substantial performance gain, highlighting the importance of explicit structural information for coordinate generation. When predicted trees are used at inference time, performance degrades gracefully, indicating that errors in structural induction propagate to geometry.

Qualitative Analysis of Tree-Conditioned Layouts

Figure 6 illustrates qualitative differences in predicted layouts. Without layout trees, global spatial relationships between panels often collapse. In contrast, tree-conditioned generation produces layouts that respect page-level partitioning and yield structurally coherent arrangements.

Figure 7 compares layouts generated using ground-truth versus predicted layout trees. Errors in structure prediction propagate directly to coordinate generation, as the model follows the predicted tree faithfully.

Failure Case Analysis

Figure 8 shows a representative failure case involving *inset panel* structures, where multiple small panels are embedded within a larger one. The current layout tree representation assigns identical structural paths to panels under the same parent node, preventing explicit modeling of containment and relative scale differences.

Beyond quantitative evaluation, the proposed framework also enables page composition by arranging generated panel images according to the predicted layout structure. An illustrative example is shown in Appendix D.2.

6 Discussion

What information supports layout structure prediction. A central finding of this study is that layout structure prediction relies much more strongly on visual representations than on content-oriented textual descriptions. The caption degradation experiment shows that weakening caption quality has a limited impact on tree edit distance, suggesting that panel content provides weak cues for recovering page-level layout structure. This indicates that layout organization is not strongly determined by narrative content alone.

This observation is further supported by analyses of visual representations. In the CLIP layer-

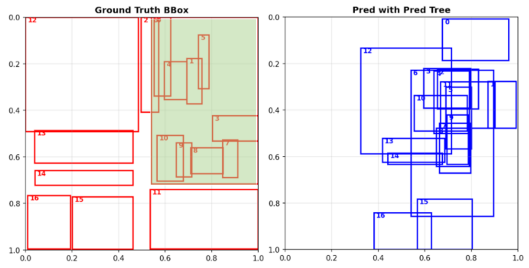

Figure 8: Failure case of an inset panel layout. Green regions mark inset panels in the ground-truth layout. Due to the presence of inset panels, the predicted layout fails to preserve the overall page structure observed in the ground truth.

wise comparison, intermediate layers achieve lower tree edit distance than higher layers, even though higher layers are stronger for semantic alignment between images and language. If semantic content were a primary factor in determining layout structure, one would expect higher-layer features to perform better; however, this trend is not observed (Figure 3). Similarly, MAE-based visual features, which are learned through masked image reconstruction rather than contrastive vision–language objectives, achieve performance comparable to or slightly better than CLIP features, despite using smaller model variants. These results suggest that the effectiveness of visual features for layout structure prediction is not primarily driven by semantic understanding of panel content.

The frequency decomposition analysis provides complementary evidence. Low-frequency image components consistently outperform high-frequency components, indicating that layout structure prediction depends more on coarse visual information than on fine-grained details such as local edges or textures. Taken together, these analyses suggest that the information supporting layout structure prediction is systematically present in certain visual representations, but is only weakly reflected in textual descriptions that focus on narrative content.

Structure as an intermediate representation for layout realization. The coordinate generation experiments clarify the role of layout structure in page-level layout modeling. Directly predicting panel coordinates from visual or textual features alone results in limited geometric accuracy, suggesting that raw content representations are insufficient for reliable coordinate regression. In contrast, conditioning coordinate generation on an explicit layout tree substantially improves mIoU, highlighting the importance of an intermediate structural representation.

These results indicate that layout trees provide an effective interface between perceptual input and geometric realization. By explicitly representing how a page is recursively partitioned, the layout tree constrains global panel organization while allowing local geometric refinement during coordinate prediction. The comparison between ground-truth and predicted trees further shows that errors in structure prediction propagate systematically to geometry, underscoring the tight coupling between structural induction and coordinate generation.

Implications for multimodal manga understanding. Our findings have implications for multimodal modeling of manga and other visual narrative media. While textual descriptions are effective for conveying narrative content, they appear insufficient for capturing page-level layout structure, which plays an important role in how visual narratives are organized and read. For layout-centric tasks in manga, representations that preserve spatial organization and structural relations are therefore also crucial, in addition to representations optimized for semantic correspondence.

7 Conclusion

This paper studied manga page layout as a key component of manga understanding. We explored whether page-level layout structure can be inferred from panel content representations and how this structure supports geometric layout generation.

Our experiments show that layout structure prediction relies much more on visual features than on content-oriented textual descriptions. Analyses, including caption degradation, visual encoder variants, and frequency decomposition, reveal that layout is weakly reflected in textual descriptions but captured by visual representations.

We also demonstrated that explicit layout structure serves as an effective intermediate representation for coordinate generation. While direct regression from content features is unreliable, conditioning on a predicted layout tree significantly improves layout realization.

Overall, our findings indicate that layout structure is a crucial intermediate representation for manga page modeling and is primarily encoded in visual features.

8 Limitations & Ethics

Limitations in Textual Representation Our analysis of textual inputs primarily focuses on content-oriented panel descriptions, which summarize what happens within each panel. While this approach reflects common captioning pipelines, it does not exhaust the space of possible textual representations. Textual descriptions that explicitly encode spatial relations, layout cues, or authorial intent may provide stronger signals for layout structure prediction, and we leave the exploration of such representations to future work.

Limitations in Layout Tree Representation The layout tree representation adopted in this study has limited expressive power. In particular, it cannot explicitly represent inset or nested panel structures, leading to characteristic failure cases in coordinate generation. Extending the representation to model containment, relative scale, or more complex panel relations remains an open challenge.

Generalization to Other Comic Styles Our experiments are conducted on the Manga109 dataset, which reflects the conventions of manga—Japanese comics. Whether the observed findings generalize to other comic styles, such as Western comics or web-based formats with different layout conventions, requires further investigation.

Limitations in Evaluation Methodology Finally, while we demonstrate an end-to-end application by composing generated panel images into full pages, this demonstration is qualitative and not systematically evaluated. A more comprehensive assessment of page-level generation quality and narrative coherence is beyond the scope of this work.

Ethics We strictly adhere to the usage guidelines set forth by the Manga109 dataset, ensuring that the manga used in this work is for academic purposes only, with proper attribution to the authors. We also do not redistribute the dataset to third parties and comply with all terms of the usage agreement. Furthermore, AI tools, such as those for translation and writing assistance, have been employed to aid in the composition and refinement of this paper. We acknowledge these contributions transparently to ensure the integrity of the research process.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. 2020. Building a manga dataset "manga109" with annotations for multimedia applications. *IEEE MultiMedia*, 27(2):8–18.

Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. 2022. Coo: Comic onomatopoeia dataset for recognizing arbitrary or truncated texts. In *European Conference on Computer Vision*, pages 267–283. Springer.

Ying Cao, Antoni B Chan, and Rynson WH Lau. 2012. Automatic stylistic manga layout. *ACM Transactions on Graphics (TOG)*, 31(6):1–10.

Siyu Chen, Dengjie Li, Zenghao Bao, Yao Zhou, Lingfeng Tan, Yujie Zhong, and Zheng Zhao. 2024. Manga generation via layout-controllable diffusion. *arXiv preprint arXiv:2412.19303*.

Neil Cohn. 2013. *The Visual Language of Comics: Introduction to the Structure and Cognition of Sequential Images*. Bloomsbury.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th annual ACM symposium on user interface software and technology*, pages 845–854.

David Dubray and Jochen Laubrock. 2019. Deep cnn-based speech balloon detection and segmentation for comic books. *CoRR*, abs/1902.08137.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338.

Siyuan Feng, Teruya Yoshinaga, Katsuhiko Hayashi, Koki Washio, and Hidetaka Kamigaito. 2025. How panel layouts define manga: Insights from visual ablation experiments. In *Proceedings of the 47th Annual Meeting of the Cognitive Science Society (CogSci 2025)*, Rotterdam, Netherlands. Cognitive Science Society.

Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. 2023. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in Neural Information Processing Systems*, 36:18225–18250.

Kamal Gupta, Justin Lazarow, Alessandro Achille, Larry S Davis, Vijay Mahadevan, and Abhinav Shrivastava. 2021. Layouttransformer: Layout generation and completion with self-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1004–1014.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009.

HsiaoYuan Hsu and Yuxin Peng. 2025. Postero: Structuring layout trees to enable language models in generalized content-aware layout generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8117–8127.

Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. Docbank: A benchmark dataset for document layout analysis. *arXiv preprint arXiv:2006.01038*.

Yingxuan Li, Ryota Hinami, Kiyoharu Aizawa, and Yusuke Matsui. 2024. Zero-shot character identification and speaker prediction in comics via iterative multimodal fusion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7366–7374.

Yonggang Li, Yafeng Zhou, Yongtao Wang, Xiaoran Qin, and Zhi Tang. 2021. Dual loss for manga character recognition with imbalanced training data. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2166–2171. IEEE.

Jiadong Liang, Wenjie Pei, and Feng Lu. 2023. Layout-bridging text-to-image synthesis. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(12):7438–7451.

Jiawei Lin, Jiaqi Guo, Shizhao Sun, Weijiang Xu, Ting Liu, Jian-Guang Lou, and Dongmei Zhang. 2023. A parse-then-place approach for generating graphic layouts from textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23622–23631.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2017. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838.

Scott McCloud and Mark Martin. 1993. *Understanding comics: The invisible art*, volume 106. Kitchen sink press Northampton, MA.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2023. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Christophe Rigaud, Nhu-Van Nguyen, and Jean-Christophe Burie. 2021. Text block segmentation in comic speech bubbles. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part VI*, pages 250–261. Springer-Verlag.

Ragav Sachdeva, Gyungin Shin, and Andrew Zisserman. 2024. Tails tell tales: Chapter-wide manga transcriptions with character names. In *Proceedings of the Asian Conference on Computer Vision*, pages 2053–2069.

Ragav Sachdeva and Andrew Zisserman. 2024. The manga whisperer: Automatically generating transcriptions for comics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12967–12976.

Ragav Sachdeva and Andrew Zisserman. 2025. From panels to prose: Generating literary narratives from comics. *arXiv preprint arXiv:2503.23344*.

Hengyu Shi, Junhao Su, Huansheng Ning, Xiaoming Wei, and Jialin Gao. 2025. Layoutcot: Unleashing the deep reasoning potential of large language models for layout generation. *arXiv preprint arXiv:2504.10829*.

Haruka Takahashi and Shigeru Kuriyama. 2024. Text2layout: Layout generation from text representation using transformer. *IEEE Access*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Jianzong Wu, Chao Tang, Jingbo Wang, Yanhong Zeng, Xiangtai Li, and Yunhai Tong. 2025. Diffsensei: Bridging multi-modal llms and diffusion models for customized manga generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28684–28693.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Tao Yang, Yingmin Luo, Zhongang Qi, Yang Wu, Ying Shan, and Chang Wen Chen. 2024. Posterllava: Constructing a unified multi-modal layout generator with llm. *arXiv preprint arXiv:2406.02884*.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, 18(6):1245–1262.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: largest dataset ever for document layout analysis. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 1015–1022. IEEE.

A Dataset Details

A.1 Manga109Caption Construction

We construct **Manga109Caption** by augmenting the Manga109 dataset with panel-level captions. Each caption describes the visual content of a panel and is generated automatically using LLaVA-v1.6-34B. No manual correction or post-editing is applied, ensuring that the captions reflect local panel content without encoding page-level layout information. These captions are used solely to derive textual input features and do not contribute to any language generation component.

A.2 Caption Prompting and Degradation Strategy

To investigate the role of textual descriptions in layout prediction, we generate captions under two prompting conditions: normal and degraded.

Normal prompt: *"Describe the scene in one concise English sentence. Focus on the actions of the characters, their surroundings, and the overall atmosphere, without unnecessary details."*

Degraded prompt: *"Describe this panel from a Japanese manga."*

The degraded prompt tests whether the limited performance of text-based models arises from caption quality or from a fundamental mismatch between content-oriented descriptions and layout structure prediction.

Specific examples of captions generated under the two conditions are illustrated in Figure 9. The caption generated by the normal prompt includes more context about the scene, including actions and atmosphere, while the degraded prompt provides only a brief description of the main elements.

B Layout Tree Construction

In constructing the layout tree of a manga page, the main goal is to recursively partition the page into distinct regions based on the panel bounding box coordinates. This hierarchical structure is created through a series of divisions based on the spatial relationships between panels.

First, the page is divided into two major sections: the RIGHT and LEFT regions, based on the centerline of the page. Panels are recursively subdivided based on their spatial relationships, first along rows and then along columns, depending on their relative positions and overlap. This process generates a layout tree that reflects the spatial organization of the panels.

Inset panels, which are small panels contained within larger ones, inherit the layout path of their parent panel, ensuring that the hierarchy is maintained. Each panel's layout path is recorded, which allows the tree structure to be traversed and used for coordinate generation.

C Model and Training Details

C.1 Model Architecture Diagram

Figure 10 shows the model architecture, detailing the flow from input panels to bounding boxes via the two-stage framework. The first stage predicts a hierarchical layout tree, and the second generates the panel coordinates.

The Base BBox Prediction stage takes two key inputs: Tree Structure and Leaf Paths. Tree Structure encodes the panel's position in the layout tree as a three-dimensional vector, representing its depth, sibling index, and sibling count. Leaf Paths are strings indicating the panel's exact location in the tree, e.g., Root-RIGHT-R1-C2 for a panel in the second column of the first row within the right section. These inputs are passed into the transformer encoder, which models the global structure and spatial relationships to generate accurate coordinates.

C.2 Model Hyperparameters

Table 3 and table 4 the hyperparameters used for the layout tree prediction model and panel coordinate generation model.

Table 3: Hyperparameters for Layout Tree Prediction Model.

Hyperparameter	Default Value
Hidden dimension	256
Encoder layers	2
Decoder layers	4
Attention heads	8
FFN dimension	1024
Dropout	0.1

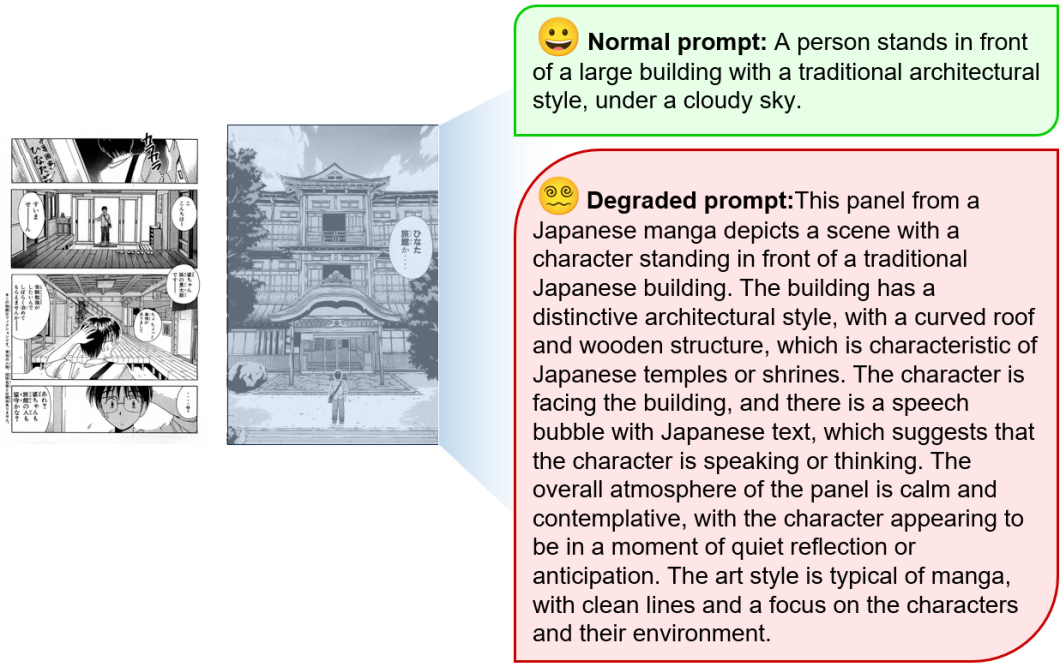

Figure 9: Example of captions generated from normal and degraded prompts. The normal prompt provides a concise and accurate description, while the degraded prompt gives an overly detailed description, which can lead to the inclusion of incorrect or unnecessary information. (Love Hina © Ken Akamatsu)

Table 4: Hyperparameters for Panel Coordinate Generation Model.

Hyperparameter	Default Value
Structure encoder layers	4
Content encoder layers	2
Hidden dimension	256
Attention heads (structure)	8
Attention heads (content)	4
FFN dimension (structure)	1024
FFN dimension (content)	512
Path encoder layers	2

C.3 Training Details

For the Panel Coordinate Generation model, the training objective is based on a regression loss using intersection-over-union (IoU), augmented with auxiliary losses. The full objective function is given by Equation 6, where the loss weights used during training are: $\lambda_{\text{iou}} = 1.0$, $\lambda_{\text{l1}} = 0.5$, and $\lambda_{\text{res}} = 0.5$. Both models use the AdamW optimizer with an initial learning rate of $1e-4$. The learning rate for the layout tree generation model is decayed using a weight decay of $1e-5$, while for the bounding box adjustment model, weight decay is set to 0.01. Gradient clipping is applied with a threshold of 1.0. Early stopping is implemented with a patience of 15 epochs and a minimum improvement threshold of 0.001. All experiments are conducted using a random seed of 42.

C.4 BiLSTM vs Transformer Encoder Comparison

In the main experiment, the Layout Tree Prediction task used a BiLSTM encoder. However, we also tested the performance of a Transformer encoder for comparison. The BiLSTM encoder has approximately 3.0M parameters, whereas the Transformer encoder has around 4.5M parameters. Despite the Transformer encoder having more parameters, it showed a slight decrease in performance compared to the BiLSTM encoder.

Table 5 summarizes the results of our comparison between the two encoders under various experimental conditions:

We also tested the impact of CLIP layer-wise features on the layout tree prediction, as shown in Figure 11.

As shown in Table 5 and Figure 11, while the Transformer encoder has more parameters, it performed slightly worse than the BiLSTM encoder across most settings. This suggests that the BiLSTM encoder, with its smaller architecture, might be better suited for this task.

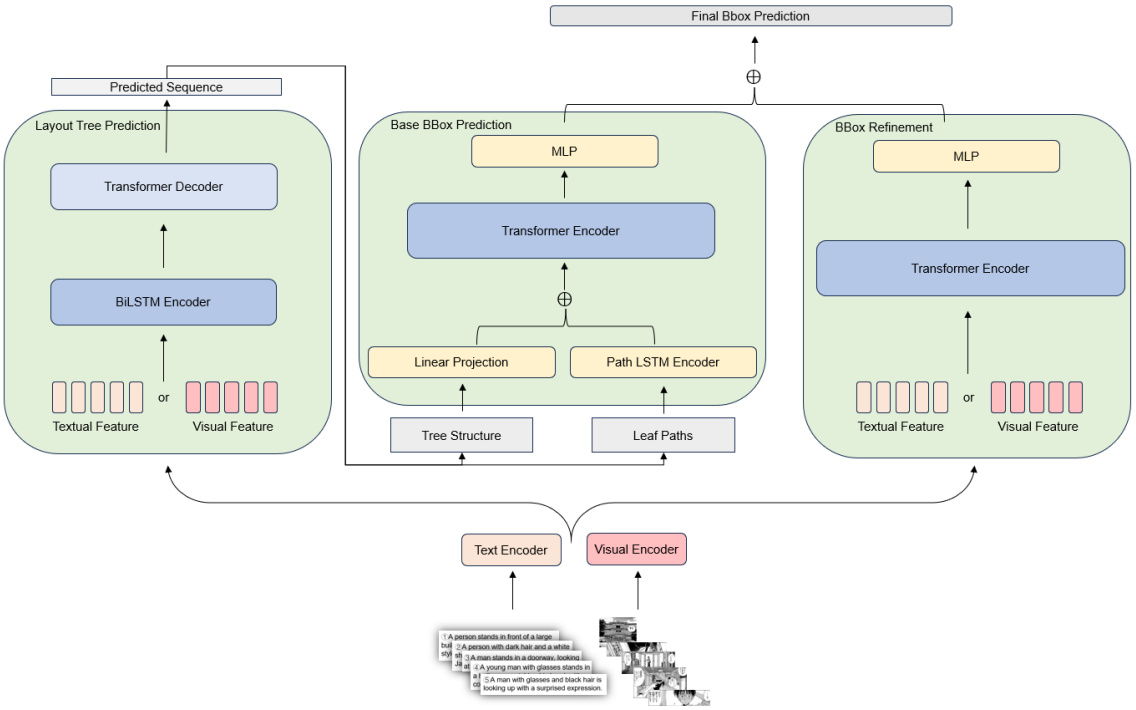

Figure 10: Detailed architecture of the proposed two-stage framework. Left: layout tree prediction. Middle: tree-conditioned base bounding box prediction. Right: coordinate refinement.

Table 5: Comparison of BiLSTM and Transformer Encoder for Layout Tree Prediction. The values represent TED results.

Setting	BiLSTM	Transformer
Text(Base)	0.33	0.34
Visual(Base)	0.21	0.22
CLIP ViT-L/14	0.20	0.22
MAE ViT-B/16	0.20	0.20
MAE ViT-L/16	0.18	0.19
High-frequency	0.19	0.20
Low-frequency	0.15	0.15

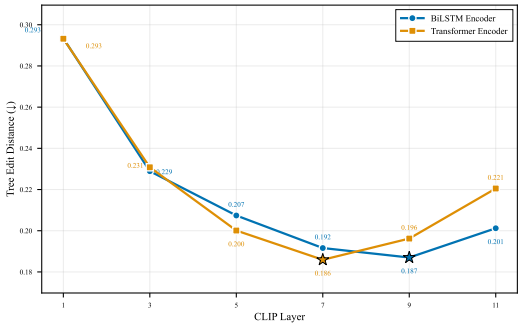

Figure 11: Comparison of TED performance using BiLSTM and Transformer encoders across different layers of the CLIP model for layout tree prediction.

D Additional Experimental Results

D.1 Additional Qualitative Results on Manga109

Figure 12, Figure 13, and Figure 14 illustrate the application of the proposed manga layout generation method on the Manga109Caption test set. These figures show two examples each from the top 20, middle 20, and bottom 20 test instances with 6 or more panels per page, selected based on their mIoU scores. They demonstrate the framework's performance across different levels of accuracy in layout generation.

D.2 Examples of Single Page Manga from Generated Panels

Figure 15 presents a single-page manga created by composing panels generated using BluePencilXL 3.10, based on the Stable Diffusion XL (SDXL) architecture (Podell et al., 2023). It demonstrates the application of our framework to synthetic panel images. While not quantitatively evaluated, it highlights the framework's ability to generate structured manga layouts. The figure shows that our layout generation method can be effectively combined with existing image generation models to create single-page manga.

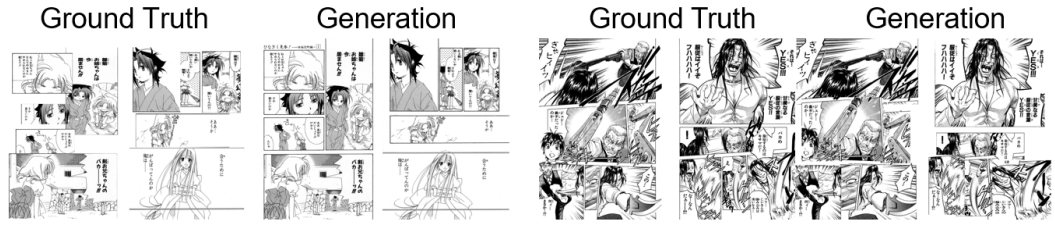

Figure 12: Two examples from the top 20 cases with the highest mIoU. The layout tree prediction results of these examples match the ground truth exactly. As seen in these cases, the predicted layouts are very close to the ground truth, confirming the effectiveness of our layout tree generation method. (Right: HinagikuKenzan ©Minene Sakurano, Left: DollGun ©Tatsumasa Deguchi)

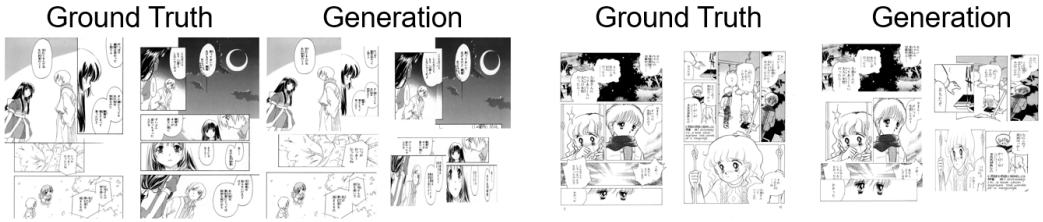

Figure 13: Two examples from the middle 20 cases with moderate mIoU. The layout tree prediction results of these examples do not match the ground truth exactly, but the errors are relatively small. As seen in these cases, the predicted layouts show some differences from the ground truth, but the overall panel structure is still recognizable. (Right: HinagikuKenzan ©Minene Sakurano, Left: SonokiDeABC ©Kimi Takishiro)

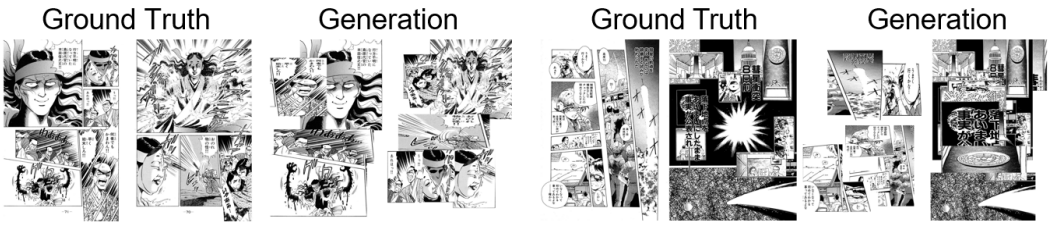

Figure 14: Two examples from the bottom 20 cases with the lowest mIoU. The layout tree of the left example deviates significantly from the ground truth, resulting in a low mIoU, but the overall manga panel structure is still recognizable. The right example is a typical inset panel case, where many small panels are embedded within a large panel in the right half of the page. The current layout tree representation treats the paths of these panels as identical, unable to capture the inherent relationship, leading to very poor coordinate prediction results. (Right: YoumaKourin ©Jo Shimazaki&Tsukasa Takatsu, Left: EvaLady ©Shii Gomi)

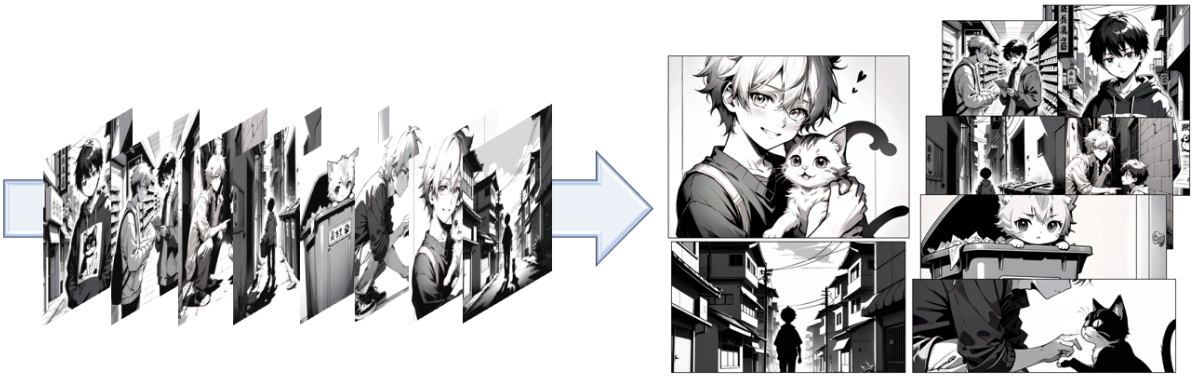

Figure 15: A single-page manga generated using BluePencilXL 3.10 based on the Stable Diffusion XL (SDXL) architecture. The image illustrates how our layout generation framework can be applied to synthetic panel images for creating structured manga layouts.