Advancing Manga Analysis: Comprehensive Segmentation Annotations for the Manga109 Dataset - Supplementary Material

Minshan Xie[1], Jian Lin[2], Hanyuan Liu, Chengze Li[2], Tien-Tsin Wong[3]
[1]Centre for Perceptual and Interactive Intelligence, Hong Kong SAR,
[2]Saint Francis University, Hong Kong SAR [3]Monash University, Australia
msxie92@gmail.com {jlin,czli}@sfu.edu.hk drhanyuanliu@gmail.com TT.Wong@monash.edu

A. Data and Code Availability

The code and our **MangaSeg** annotation are publicly available at https://huggingface.co/datasets/MS92/MangaSegmentation. Our annotation is released under a customized license that shares the same spirit as the Creative Commons Attribution 4.0 International License (CC BY 4.0). The existing annotation is currently self-contained. However, we have committed to ongoing updates for at least one or two years. If any issues arise with the annotation (such as inaccuracies, missing annotations, or other anomalies), we will promptly address them to maintain data quality.

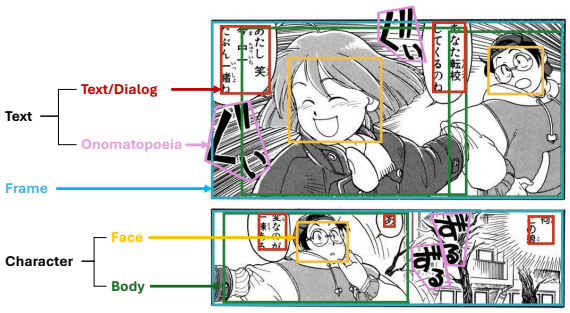
B. Manga109 Dataset

Before getting into the details of our segmentation annotation, we need to briefly introduce Manga109 and their bounding box annotations. The Manga109 dataset is a collection of Japanese comic books, consisting of 109 volumes created by 94 professional manga authors [8]. Permission to use the Manga109 dataset is granted by the author of each work included in the dataset on the condition that the dataset will be used for academic purposes in non-profit organizations, for research-related purposes such as experimentation and publication of academic papers.

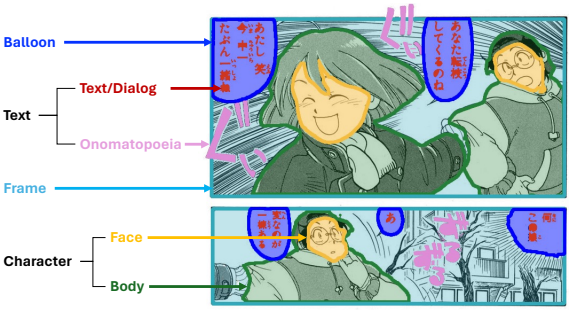
B.1. Manga109 Annotations

The dataset includes annotations for frames, speech texts, character faces, and character bodies, with over 500,000 total annotations [1] (Fig. 1(a)). Each volume of manga corresponds to a single xml file. Each page gives overview (page number, size of the image) and objects information in the page. There are four categories of objects. For each object, object ID, its rectangular area (xmin, xmax, ymin, ymax), and additional information specific to the object type are given. They also provide manga109api https://github.com/manga109/manga109api, which is a simple python API to read annotation data. For more details, please refer to their official website http://www.manga109.org/en/annotations.html.

There are some additional annotations in Manga109. Character gives the correspondence between each character's name and its ID while text also label the text content. The text is further extended with speaker-to-text annotation



(a) Bounding box annotations



(b) Segmentation annotations

Figure 1. (a) Manga109 defines four different bounding box annotations (Frame, Text, Character Face and Character Body) and content annotations (Text Content and Character Name). (b) In our annotation, we define six object categories: frame, text/dialog, onomatopoeia, character body, character face, and balloon. For better visualization, we highlight their boundaries in the figure. AppareKappore ©Kanno Hirosh [1]

[4]. They also provide the COmic Onomatopoeia annotation (COO), i.e. onomatopoeia text [2].

C. MangaSeg Annotations

As redistribution of any part of the Manga109 dataset to third parties is forbidden, we only provide our MangaSeg annotation at https://huggingface.co/datasets/MS92/MangaSegmentation. We also provide code to access the content annotations in the Manga109 dataset.

C.1. Data Structure

In our annotation, we define six object categories: frame, text/dialog, onomatopoeia, character body, character face, and balloon, as shown in Figure 1(b). We create our dataset with COCO-style JSON annotation. Our annotations are provided in JSON format, with each file corresponding to each volume (Listing 1). Each annotation contains "info," "licenses," "images," "annotations" and "categories" fields. The "images" field contains a list of images of the volume with "id," "width," "height" and "file_name." The "annotations" field contains a list of annotations detected in the images. Each annotation element in the list includes "id," "image_id," "category_id," "bbox," "area," and "segmentation" fields. The "segmentation" field is RLE format for segmentation masks. The "categories" field defines a unique ID for each of the six categories of manga components, i.e. "frame": 1, "text": 2, "face": 3, "body": 4, "balloon": 5, "onomatopoeia": 6. We also provide example code to visualize our annotations.

D. Dataset Motivation and Intended Uses

The original bounding box annotations in Manga109 dataset is not sufficient for many manga-related applications, such as manga inpainting [15], manga localization [14, 17] and manga retargeting [16]. In contrast, our dataset creation pipeline is designed for generating training data that suits for deep learning-based instance segmentation models of manga images. It allows us to identify individual components within a manga page. By segmenting these components, we shall be able to achieve more accurate analysis (both geometry and semantic) of the manga.

One of the most promising applications of this technique is the automatic digital migration of manga using text segmentation and onomatopoeia segmentation [17]. The character body segmentation effectively separates characters from the background, making it easier to animate them or integrate them into different scenes. The character face segmentation can help to better identify the facial expression and even produce simple speaking animation (probably with lip-sync). The balloon segmentation can facilitate the background inpainting tasks and motion manga. By leveraging with the detailed content annotations in Manga109 (as described by [1]), such as character identification, we can explore additional use cases, such as instance-level manga colorization.

Beyond its immediate benefits, our annotation opens up groundbreaking possibilities for the research community. The impact of manga instance segmentation extends beyond creators and editors. It enriches the manga reading experience by enabling dynamic motion (through image-to-video generative model) and interactive features. We enthusiastically invite contributions from the community to further explore the potential applications of our segmentation annotation.

E. Benchmark results

E.1. Data split of the dataset

The following per domain split of the data has been used for the experiments. In this work, we use the Pytorch toolkit [10] to conduct all experiments on four nVidia GeForce RTX 3090 GPUs with a batch size of 8. We fine-tune the SAM mask decoder model with additional instance tokens, which includes fine-tuning the image encoder as well [5]. Training spans 100 epochs using the AdamW [7] optimizer with a learning rate of 0.002.

E.2. Experimental Results

More qualitative examples can be found in Figures 4, 3, 5, 6, 7. To enhance visualization, we overlay the detected masks onto the original manga images. Our results demonstrate that fine-tuning our model with an augmented dataset significantly improves its performance compared to the original SAM model. The distinctive features of manga in our augmented dataset improve the ability of the model to handle exaggerated drawing styles and the black-and-white screentone filled characteristics more effectively. GroundedSAM [11] may struggle with understanding black-and-white manga images due to domain gaps. For non-learning-based methods, e.g. [6] for balloon extraction and [9] for frame/panel extraction, our fine-tuned model exhibits significant improvement and remains robust even in the presence of unclosed regions. Regarding text and body extraction, models trained on datasets with paired manga images and corresponding labeling masks, such as [3] and [5], perform better than GroundedSAM [11]. This highlights the critical role of segmentation annotation in the dataset.

References

[1] 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. 1, 2

```
{
    "info": {
        "year": 2024,
        "version": "1.0",
        "description": "Manga109 Segmentation",
        "contributor": "Minshan XIE",
        "url": "https://huggingface.co/datasets/MS92/MangaSegmentation",
    },
    "licenses": [
       {
         "id": 1,
         "name": "Attribution License",
         "url": "http://creativecommons.org/licenses/by/4.0/"
       }
    ],
     "images": [
         {
           "license": 0,
           "id": 0,
           "width": 1654,
           "height": 1170,
           "file_name": "ARMS/000.jpg"
       },
         // More image entries here...
     ],

     "categories": [
       {"id": 1, "name": "frame", "supercategory": "frame"},
       {"id": 2, "name": "text", "supercategory": "text"},
       {"id": 3, "name": "face", "supercategory": "character"},
       {"id": 4, "name": "body", "supercategory": "character"},
       {"id": 5, "name": "balloon", "supercategory": "balloon"},
       {"id": 6, "name": "onomatopoeia", "supercategory": "text"}
     ],
     "annotations": [
         {
             "id": 0,
             "image_id": 2,
             "category_id": 1,
             "segmentation": [RLE format],
             "area": 670660,
             "bbox": [x, y, width, height],
             "iscrowd": 0
         },
         // More annotation entries here...
     ]
}
```

Listing 1. **Data Structure.** Our segmentation annotation is saved with COCO-style JSON format.

[2] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. Coo: Comic onomatopoeia dataset for recognizing arbitrary or truncated texts. In *ECCV*, 2022. 2

[3] juvian. Manga&comic text detection. GitHub repository, 2023. 2

[4] Yingxuan Li, Kiyoharu Aizawa, and Yusuke Matsui. Manga109dialog: A large-scale dialogue dataset for comics speaker detection. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2024. 2

[5] Jian Lin, Chengze Li, Xueting Liu, and Zhongping Ge. Instance-guided cartoon editing with a large-scale dataset. *arXiv preprint arXiv:2312.01943*, 2023. 2, 6

[6] Xueting Liu, Chengze Li, Haichao Zhu, Tien-Tsin Wong, and Xuemiao Xu. Text-aware balloon extraction from manga. *The Vis. Comput.*, 32(4):501–511, 2016. 2, 5, 7

[7] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[8] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia tools and applications*, 76:21811–21838, 2017. 1, 5, 6, 7

```python
import json
import os
import pycocotools.mask as maskUtils
from PIL import Image
import numpy as np
from collections import defaultdict

rle2mask = maskUtils.decode

def json2dict(json_path: str):
    with open(json_path, 'r', encoding='utf8') as f:
        metadata = json.loads(f.read())
    return metadata

json_dict = json2dict(r'json/ARMS.json')
annotations_point = json_dict['annotations']

img_annotation_dict = defaultdict(list)
for ann in annotations_point:
    img_annotation_dict[ann['image_id']].append(ann)

image_file = json_dict['images']
img_dict = {img_file['id']: img_file for img_file in image_file}

img_id = 4
cat_id = 1

img_info = img_dict[img_id]
visualize = np.zeros((img_info['height'], img_info['width'], 3), dtype=np.uint8)
annotations = img_annotation_dict[img_id]

for ann in annotations:
    if ann['category_id'] == cat_id:
        mask = rle2mask(ann['segmentation'])
        visualize[mask > 0] = np.random.randint(0, 255, (1, 3))

if visualize is not None:
    visualize = Image.fromarray(visualize)
    os.makedirs(os.path.dirname(img_info['file_name']))
    visualize.save(img_info['file_name'])
```

Figure 2. We provide sample code to visualize segmentation masks for each image.

[9] Xufang Pang, Ying Cao, Rynson W. H. Lau, and Antoni B. Chan. A robust panel extraction method for manga. In *ACM MM*, pages 1001–1004, 2014. 2, 5

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 32, 2019. 2

[11] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 2, 5, 6, 7

[12] Ragav Sachdeva and Andrew Zisserman. The manga whisperer: Automatically generating transcriptions for comics. In *CVPR*, pages 12967–12976, 2024. 5, 6, 7

[13] Baris Batuhan Topal, Deniz Yuret, Tevfik Metin Sezgin, et al. Domain-adaptive self-supervised face & body detection in drawings. In *IJCAI*, pages 1432–1439, 2023. 6

[14] Minshan Xie, Chengze Li, Xueting Liu, and Tien-Tsin Wong. Manga filling style conversion with screentone variational autoencoder. *ACM TOG*, 39(6):1–15, 2020. 2

[15] Minshan Xie, Menghan Xia, Xueting Liu, Chengze Li, and Tien-Tsin Wong. Seamless manga inpainting with semantics awareness. *ACM TOG*, 2021. 2

[16] Minshan Xie, Menghan Xia, Xueting Liu, and Tien-Tsin Wong. Screentone-preserved manga retargeting. In *Eurographics*, 2025. 2

[17] zyddnys. Manga image translator. https://github.com/zyddnys/manga-image-translator, 2025. 2

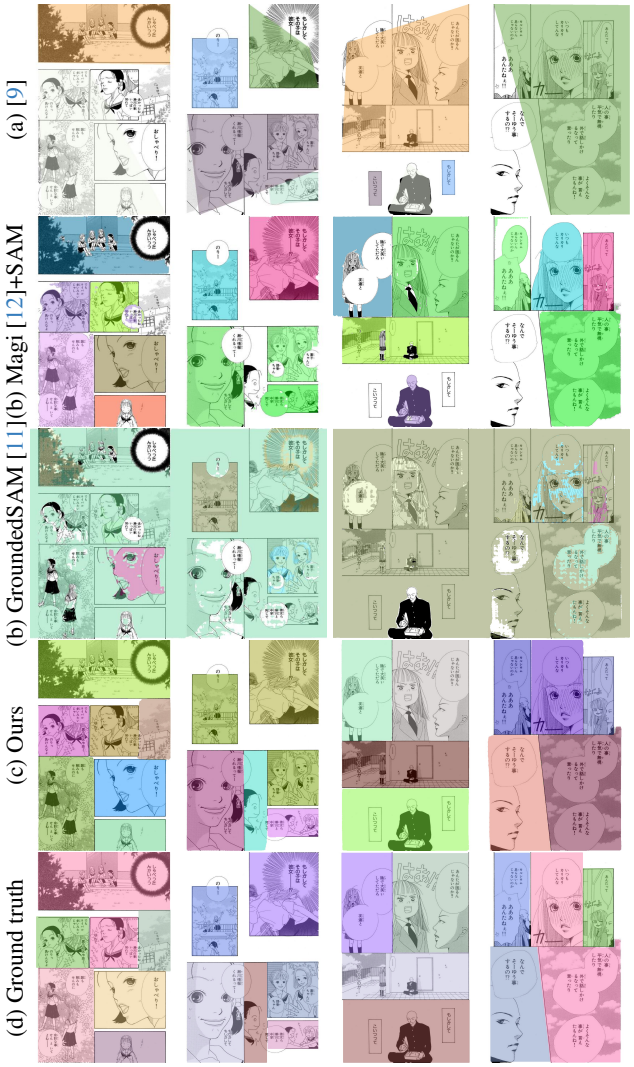


Figure 3. Frame segmentation using existing methods. YukiNoFuruMachi ©Yamada Uduki [8]

Figure 4. Balloon segmentation using existing methods. YukiNoFuruMachi ©Yamada Uduki [8]

Figure 5. Character body segmentation using existing methods. YamatoNoHane ©Saki Kaori [8]

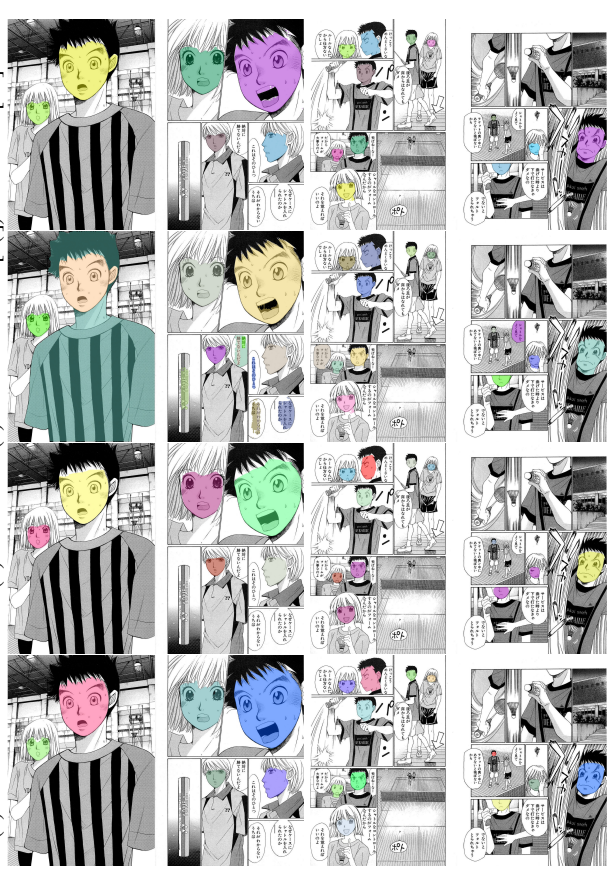


Figure 6. Character face segmentation using existing methods. YamatoNoHane ©Saki Kaori [8]

Figure 7. Text segmentation using existing methods. UltraEleven ©Yabuno Tenya, Watanabe Tatsuya [8]