The Manga Whisperer: Automatically Generating Transcriptions for Comics

Ragav Sachdeva Andrew Zisserman

Visual Geometry Group, Dept. of Engineering Science, University of Oxford

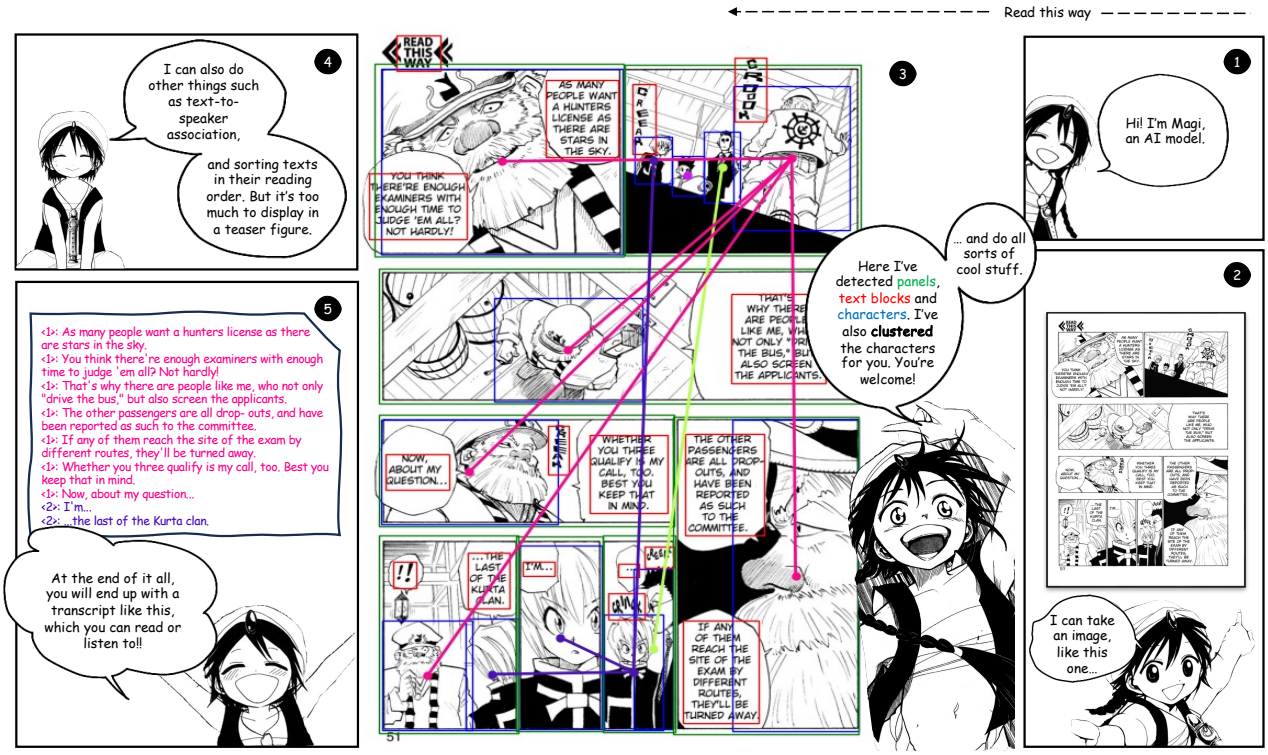

Figure 1. Given a manga page, our model is able to: (a) detect panels, text blocks and character boxes; (b) cluster character boxes by their identity; (c) associate texts to their speaker; and (d) generate a dialogue transcription in the correct reading order. Here we show the predicted panels (in green), text blocks (in red) and characters (in blue) on a page from *Hunter* × *Hunter* by Yoshihiro Togashi. The predicted character identity associations are shown by lines joining the character box centres. For visual clarity, we do not explicitly show the text to speaker associations but provide the generated transcript.

Abstract

In the past few decades, Japanese comics, commonly referred to as Manga, have transcended both cultural and linguistic boundaries to become a true worldwide sensation. Yet, the inherent reliance on visual cues and illustration within manga renders it largely inaccessible to individuals with visual impairments. In this work, we seek to address this substantial barrier, with the aim of ensuring that manga can be appreciated and actively engaged by everyone. Specifically, we tackle the problem of diarisation i.e. generating a transcription of who said what and when, in a fully automatic way.

To this end, we make the following contributions: (1) we present a unified model, ***Magi****, that is able to (a) detect panels, text boxes and character boxes, (b) cluster characters by identity (without knowing the number of clusters apriori), and (c) associate dialogues to their speakers; (2) we propose a novel approach that is able to sort the detected text boxes in their reading order and generate a dialogue transcript; (3) we annotate an evaluation benchmark for this task using publicly available [English] manga pages. The code, evaluation datasets and the pre-trained model can be found at:* https://github.com/ragavsachdeva/magi.

1. Introduction

From billboards in New York's Times Square to murals in Paris, manga characters, in all their colourful splendor, are everywhere. It is undeniably evident that manga's ubiquity extends far and wide, transcending cultural and geographic boundaries to become a cherished and adored art form, captivating the hearts of enthusiasts from diverse walks of life and backgrounds across the globe. Manga's popularity can be attributed to its rich and diverse content that covers an extensive range of genres and themes, catering to a broad and varied audience. From action-packed *shounen* manga to the intricate storytelling of *seinen* manga, the depth and breadth of storytelling in manga is unparalleled. These diverse set of themes allow readers to connect with a manga on a personal level, making it a universal form of entertainment. There is truly a manga for everyone... except for people with visual impairments (PVI).

The interest of PVI to be able to access comics is well documented [7, 23, 30, 39, 43]. In a recent study [19], conducted to understand the accessibility issues that PVI experience with comics, when the participants were asked to select the most important piece of information they wished to know while reading a comic, the majority responded with scene descriptions, followed by transcriptions, facial expressions of characters, etc. With the recent advances in computer vision and deep learning the time is right to attempt to extract this information automatically, and this work is a step towards realising that goal.

The task of "understanding" manga fully automatically, to then describe it to PVI, is *very* challenging. This involves solving a series of problems including panel detection, panel ordering, text detection, OCR, character detection, character identification, text-to-speaker association, scene captioning, action recognition, etc. As shown in Figure 1, this is extremely challenging. Characters are often drawn with varying fidelity and can be seen from different views (front, back, profile etc.), in various poses, and possibly only partially due to occlusions from speech balloons or artistic choices. It is also quite common to have non-human characters (e.g. monsters). Furthermore, text blocks may or may not be inside speech balloons which, in turn, may or may not have tails indicating the speaker. The artistic layout, special visual effects and resolution pose additional difficulties. As humans, we are able to use context and deductive reasoning to understand these complex manga, but for machines it is still very challenging.

In this work, our objective is *diarisation* – to be able to generate a transcription, page by page, of who said what in the veridical order, to convey the story on that page. To this end, we develop a model, *Magi*, for this task using a unified architecture. In order to achieve diarisation, we must address a major portion of the challenges noted above. Specifically, we tackle the problem of detection (panel, text and characters) and association (character-character and character-text) and treat it as a graph generation problem (where detections are the "nodes" and their associations are the "edges"). This completes the *computer vision* aspect of the description. We then mobilise prior knowledge of manga layout to generate a transcription from the graph with the correct ordering. This separation into two stages (graph generation, and transcription) enables a lighter model to be used than a single model trained end-to-end for the task.

To summarise, we make the following contributions: (1) we present a novel model that ingests a manga page and is able to (a) detect panels, characters and text blocks, (b) cluster characters by their identity (without making any assumptions about the number of unique identities), and (c) associate dialogues to their speakers; (2) we propose a novel method to generate a dialogue transcript using the extracted panels, text blocks and associations; (3) we create a challenging evaluation benchmark, called *PopManga*, comprising of pages from 80+ popular manga by various artists known for their complexity and detailed story-telling, and demonstrate the superior performance of our method over prior works.

2. Related Works

The problem of comic understanding (not limited to manga) is not new. ***(i)*** There are several existing works that propose solutions for panel detection [12, 32, 33, 35, 40, 46], text/speech balloon detection [3, 32, 33, 36] and character detection [14, 15, 33, 44]. Specifically for manga text and panel detection, the most successful approaches train their models on the Manga109 [1] dataset. This dataset consists of 21000+ images and, in its most recent version, provides annotations for panels, text blocks, characters (face and body), character identification, text to speaker associations [21], onomatopoeia [3] etc. For character detection, the state-of-the-art method [44] leverages domain adaptation and transfer learning techniques to achieve impressive results. ***(ii)*** For character re-identification, [45] proposes an unsupervised approach to cluster manga character faces. [37] introduces a method that leverages transfer learning and iteratively refines labels for positives and negatives to train their model. [47] propose an unsupervised face-body clustering method that incorporates the panel index of the characters for spatial-temporal information. Recently, [42] use a SimCLR [6] style approach to train a unified model that works for both face and body character crops. ***(iii)*** For speaker identification, [40] construct a bipartite graph of characters and texts and treat speaker association as a matching problem based on the Euclidean distance. Recently, [21] released dialogue annotations for Manga109 and propose a scene-graph based approach to address the speaker identification problem. We compare our method with prior works that are available publicly.

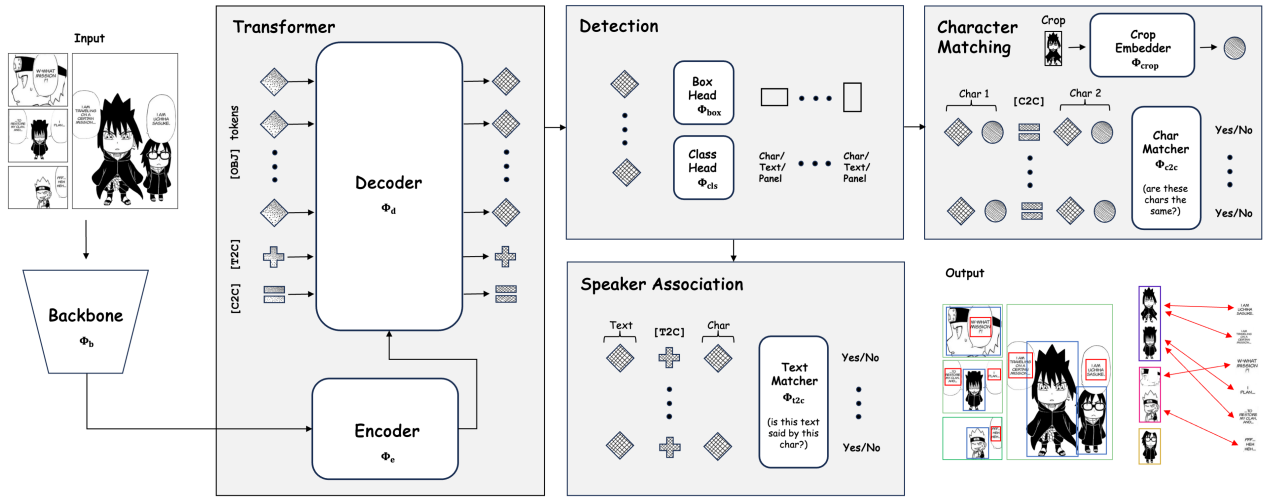

Figure 2. The *Magi* Architecture: Given a manga page as input, our model predicts bounding boxes for panels, text blocks and characters, and associates the detected character-character and text-character pairs. The model ingests a high resolution manga page as input to a CNN backbone, followed by a transformer encoder-decoder resulting in $N \times$ `[OBJ]` + `[C2C]` + `[T2C]` tokens. The `[OBJ]` tokens are processed by the detection heads (box and class) to obtain the bounding boxes and their classifications. The `[OBJ]` tokens corresponding to detected objects are then processed in pairs, along with `[C2C]` and `[T2C]`, by a character matching module and a speaker association module respectively resulting in character clusters and diarisation.

3. Detection and Association

Given a manga page, our goal is to produce a transcript of who said what and when in a fully automatic way. Therefore, as a requirement, the model must be aware of the various components that constitute a manga page—particularly the panels, characters and text blocks—and how they are related. To this end, we need to *detect* the panels, characters and text blocks (i.e. where they are on the page), as well as *associate* them – character-character association (i.e. clustering), and text-character association (i.e. speaker identification). The *detection* part is relatively straightforward; there is a plethora of methods that can be used [5, 29, 48, 49]. The tricky part is the *association*, and particularly character-character association.

Architecture overview: We formulate these tasks as graph generation problem and propose a unified model, *Magi*, that is able to simultaneously detect panels, text blocks and characters (nodes of the graph), and perform character-character matching and text-character matching (edges of the graph). The architecture is illustrated in Figure 2. The input is an entire high resolution manga page, that is processed by a CNN-based backbone to obtain a spatial feature map. These dense feature descriptors are then processed by a DETR [5] style encoder-decoder transformer, resulting in N `[OBJ]` tokens and 2 special tokens—`[C2C]`, `[T2C]`—that we introduce, inspired by [41]. The `[OBJ]` tokens are processed by the detection heads (box and class) to obtain the bounding boxes and their classifications (into character, text, panel, or background). The `[OBJ]` tokens corresponding to detected objects are then processed in pairs, along with `[C2C]` and `[T2C]`, by a character matching module and a speaker association module respectively, which essentially answer the question *"Is there an edge between a given pair of `[OBJ]` nodes?"*, resulting in character clusters and speaker associations.

Design choices: We perform "in-context" detection and association, where the network ingests the entire page. In contrast, previous methods for character re-identification [37, 42, 45, 47] operate on *crops* of characters (either faces or full-body) and use metric learning to learn desirable feature representations which can then be used to compute a similarity score given two character crops. This approach is limited in two ways: (a) metric learning based solutions are good for the retrieval task where there is query image and a large gallery of candidates to retrieve matches from, but they are not well suited to make hard decision boundaries (i.e. form clusters) especially when the number of clusters is unknown apriori and the number of data points per cluster is too few (as is the case for a single manga page); (b) only operating on character crops loses the surrounding visual cues on the page that can assist with grouping characters. In Sec. 6 we compare our model with crop-based methods and validate our design choices.

3.1. Architecture details

Backbone is a Convolutional Neural Network (CNN), represented by $\Phi_b(\cdot)$, that ingests a high resolution input

manga page $I \in \mathbb{R}^{3\times H\times W}$ and extracts dense feature descriptors $f \in \mathbb{R}^{c\times hw}$.

Transformer Encoder is a DETR-style transformer encoder, represented by $\Phi_e(\cdot)$, which processes content embeddings f from the backbone with the aim of improving them, resulting in features $g \in \mathbb{R}^{c\times hw}$.

Transformer Decoder is a DETR-style transformer decoder with conditional cross-attention [29], represented by $\Phi_d(\cdot)$. It processes $N+2$ *query* tokens as input, which consist of $N\times$`[OBJ]` + `[C2C]` + `[T2C]` tokens, that undergo a series of self-attention layers that perform interactions between the query latents, and cross-attention layers that attend to the contextualised image features g. The idea here is that the `[OBJ]` tokens learn to attend to specific spatial positions in the image, encoding the information pertaining to that region, while `[C2C]` and `[T2C]` learn to encode the interactions between these objects in a pooled fashion. The output of $\Phi_d(\cdot)$ is represented by $h^{obj} \in \mathbb{R}^{c\times N}, h^{c2c} \in \mathbb{R}^{c\times 1}, h^{t2c} \in \mathbb{R}^{c\times 1}$.

Detection Head is a two-component module that processes h^{obj} using (i) a MLP $\Phi_{box}(\cdot)$ to regress the object locations, and (ii) a linear layer $\Phi_{cls}(\cdot)$ to classify the objects.

Crop-Embedding Module is a ViT [8] model, represented by $\Phi_{crop}(\cdot)$ that, given a crop of the detected character i, outputs an embedding vector $c_i \in \mathbb{R}^{c\times 1}$.

Character Matching Module is an MLP, represented by Φ_{c2c}, followed by sigmoid activation that outputs a score A^{char}_{ij} of character i being the same as character j. It does so by processing a single feature vector that is obtained by concatenating $h^{obj}_i, c_i, h^{c2c}, h^{obj}_j, c_j$, where c_i, c_j are the crop embedding vectors for the character i, j respectively (crop is obtained using the predicted boxes). The final binary prediction of whether i and j are the same characters is obtained by thresholding A^{char}_{ij} using a hyperparamter τ.

Speaker Association Module is also a MLP, represented by Φ_{t2c}, followed by sigmoid activation that outputs a score A^{text}_{ij} of text i being said by character j. It does so by processing a single feature vector that is obtained by concatenating $h^{obj}_i, h^{t2c}, h^{obj}_j$. The final speaker prediction for text i is given by

$$\text{Speaker}_i = \arg\max_j A^{text}_i$$

3.2. Training

Achieving end-to-end training for this model, i.e. to jointly optimise the model for each of the detection and association tasks, requires a dataset that contain bounding boxes for panels, text blocks, characters, as well as annotations for character clusters and text-speaker associations, for each page. In Sec. 5 we describe how we obtain two datasets with these annotations: (1) *Mangadex-1.5M*, a large-scale (1.5M images) dataset with (possibly noisy) pseudo-annotations generated automatically; and (2) *PopManga*, a smaller dataset with 55000+ images in the dev set, and high quality human annotations. We train the model in two steps, first using Mangadex-1.5M and then using PopManga *Dev* subset, in a supervised fashion.

Implementation: The input image is resized such that the shorter side is 800px and the longer side is resized appropriately to preserve the aspect ratio but capped at 1333px. It is then projected to a spatial feature map by the backbone $\Phi_b(\cdot)$, which has the ResNet-50 [10] architecture. The dimensions of the spatial feature map depend on the resolution of the input image (e.g. for a resized 1100×800px image, the spatial map is 35×25). The spatial dimension of this feature map is flattened and the channel dimension is projected to 256 before passing it to the transformer. The transformer encoder-decoder $\Phi_e(\cdot), \Phi_d(\cdot)$ have 6 layers each, with the hidden dimension of 256, and 8 attention heads. The number of `[OBJ]` decoder queries is $N = 300$. $\Phi_b(\cdot), \Phi_e(\cdot), \Phi_d(\cdot), \Phi_{box}(\cdot), \Phi_{cls}(\cdot)$, are initialised using Conditional-DETR [29] weights. The crop-embedding module $\Phi_{crop}(\cdot)$ has 12 layers, with the hidden dimension of 768 and 12 attention heads, and is initialised using MAE [11] encoder weights. The input to $\Phi_{crop}(\cdot)$ is resized to 224×224. The character matching and speaker association modules Φ_{c2c}, Φ_{t2c} are both 3-layered MLPs initialised randomly. We set the character matching threshold $\tau = 0.65$. Our training objective for detection is the same as in [29]. We further apply Binary Cross Entropy loss to the outputs of our character matching and speaker association heads. Additionally, we apply Supervised Contrastive Loss [17] to the per-page embeddings from the crop-embedding module. We trained our model end-to-end, on 2×A40 GPUs using AdamW [25] optimizer with both learning rate and weight decay of 0.0001, and effective batch size of 16.

4. Transcript Generation

Once the bounding boxes for panels, characters and text boxes have been extracted, along with the character clusters and speaker associations, generating a transcript from them is really just an OCR + Sorting problem. To sort the text boxes into their reading order, we leverage prior knowledge about manga layout. Specifically, manga pages are read from top to bottom, *right to left* (note that the reading order is different from western comics which are read top to bottom, left to right). Given this, we order the text boxes in two steps: (a) order the panels to give the relative ordering

of text boxes belonging to different panels, (b) order the text boxes within each panel. After ordering the text boxes, we perform OCR to extract the content of the texts and finally generate the transcript using all the computed data.

Panel Ordering: Previous methods [13, 18] use "cuts" to recursively split the panels into horizontal and vertical partitions. The idea is to construct a tree by recursively splitting the panels using (1) a horizontal line, (2) a vertical line, in that order of priority, and then traverse the tree in a specific order (top panels before bottom for horizontal cuts, right panels before left for vertical cuts) to get the reading order. While this method works in most cases, it fails in the case of overlapping panels, where there is no clean way to "cut" the page. We propose an improved algorithm to sort the panels in their reading order that overcomes this issue. We very briefly describe our approach below and provide more details in the arXiv version of the paper.

Approach: The idea behind our approach is to represent the panels as a directed acyclic graph (DAG) where each directed edge represents the *relative* reading order between the connecting panels, and then apply Topological Sorting [16] to this DAG. The reason our method works more robustly than an entirely "cut" based method is because it can infer the relative order of two intersecting panels by relaxing the constraint of "strictly" above/below/left/right to "largely" above/below/left/right. By considering panels in pairs, we do not require *global* clean "cuts" to exist. In Figure 3, we show the ordering prediction using our method vs prior works.

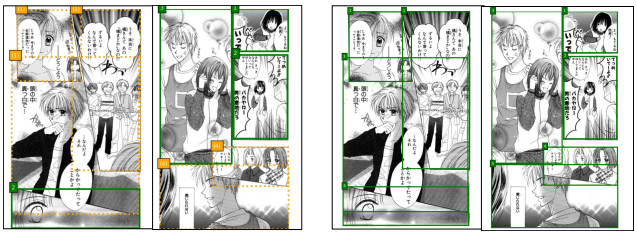

Figure 3. Panel ordering: On the left are the ordering predictions using [18] and on the right using ours. Images: Prism Heart ©Mai Asatsuki.

Intra-Panel Text Ordering: There is no absolute rule that determines the order of texts within the panel. As a general rule of thumb, however, the texts within the panel are read in the order of their distance from the top-right corner of the panel. Following previous works [13, 18], we use this heuristic to sort texts within each panel.

OCR: While performing OCR is not the primary goal of this work, we train an OCR model for completeness. Specifically, we finetune the TrOCR [20] model on custom synthetic data generated using the pipeline described in [4]. Please refer to the arXiv version for more details.

5. Datasets

For training and evaluation we require datasets that contain bounding boxes for panels, text blocks, characters, as well as annotations for character clusters and text-speaker associations, for each page. In the following we describe three datasets that cover these requirements: an existing dataset *Manga109* [1], and two new datasets that we introduce, *PopManga* and *Mangadex-1.5M*. We provide an overview of these datasets in Sec. 5.1, and how we collected and annotated the new datasets in Sec. 5.2.

5.1. Dataset overview

The three datasets are summarised in Table 1.

Dataset		Annotations					
name	#images	source	#P	#T	#C	#C2C	#T2C
PopManga (Dev)	55,393	Human	264,067†	459,615	487,367	397,054	52,011‡
PopManga (Test-S)	1,136	Human	✗	11,843	11,498	9,960	7,537
PopManga (Test-U)	789	Human	✗	9,000	7,280	5,819	6,090
Manga109 [1]	21,204	Human	104,566	147,894	157,512	167,020	131,039
Mangadex-1.5M	1.57M	Pseudo	6.6M	12M	14M	††	††

Table 1. Training and evaluation datasets. In the columns we provide the number of annotations for each task P = Panels, T = Texts, C = Characters, C2C = Character-character positive pairs, T2C = Text-character pairs. † automatically generated. ‡ for 8,488 images only. †† mined using heuristics dynamically.

PopManga: A new dataset of 57000+ manga pages from 80+ different series that are considered to be in the list the top manga of all time [31]. These manga are typically known for their complex storytelling and intricate art style, thus making them very challenging. We split PopManga into 3 subsets. The *Dev* subset is used for training and validation, and the remaining two subsets—*Test-S* (test seen), *Test-U* (test unseen)—are used for testing such that *Test-S* consists of 30 chapters from 15 series, where *other* chapters from these series are seen during training, while *Test-U* consists of 20 chapters from 10 series that the model never sees during training.

We intend the *Test-S* and *Test-U* images to serve as a public benchmark for evaluation. For this reason, they are from chapters that are available freely, publicly and officially on Manga Plus by Shueisha [28], thus enabling future academic research and comparisons. A comprehensive list of manga name and chapters in *Dev*, *Test-S*, *Test-U* is available in the arXiv version of the paper.

Manga109 [1]: An existing dataset of 21000+ images across 109 manga volumes, that provides all the detection and association annotations we require. However, in this work, we are mainly interested in the diarisation of English versions of manga which are not available in Manga109. Therefore, we only use Manga109 for some evaluation purposes.

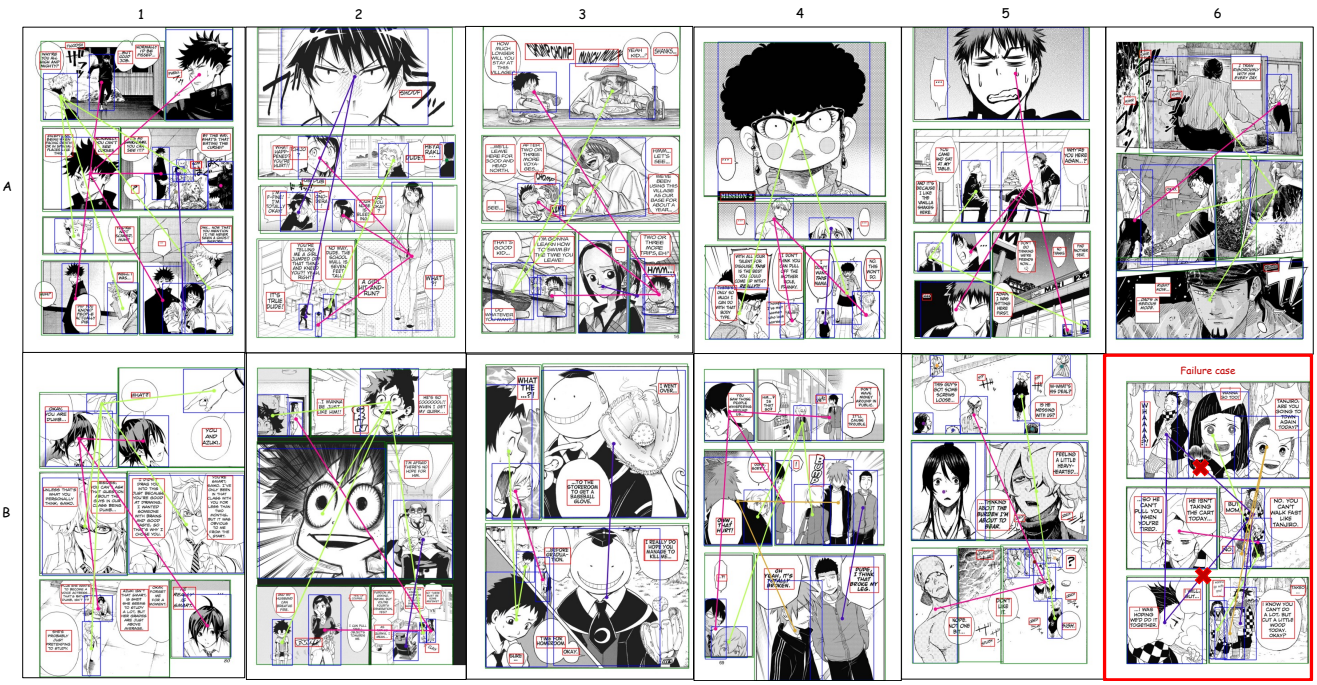

Figure 4. Bounding box predictions determined by the *Magi* model for characters, text blocks and panels, as well as clustering predictions (as nodes and edges). For the purposes of visualisation, we remove redundant connections for characters that are already connected via transitivity. Best viewed digitally. Notice that the model has successfully matched characters despite occlusion/partial visibility (girl: A4, hand: B1), changing viewpoints (boy: A5, girl: A2), and varying fidelity (girl: A1, boy: B2). The model can also detect non-human characters (dog-like creatures: A1, octopus-like creature: B3). We also show a failure case where the model incorrectly matches two different characters (one is wearing checked scarf).

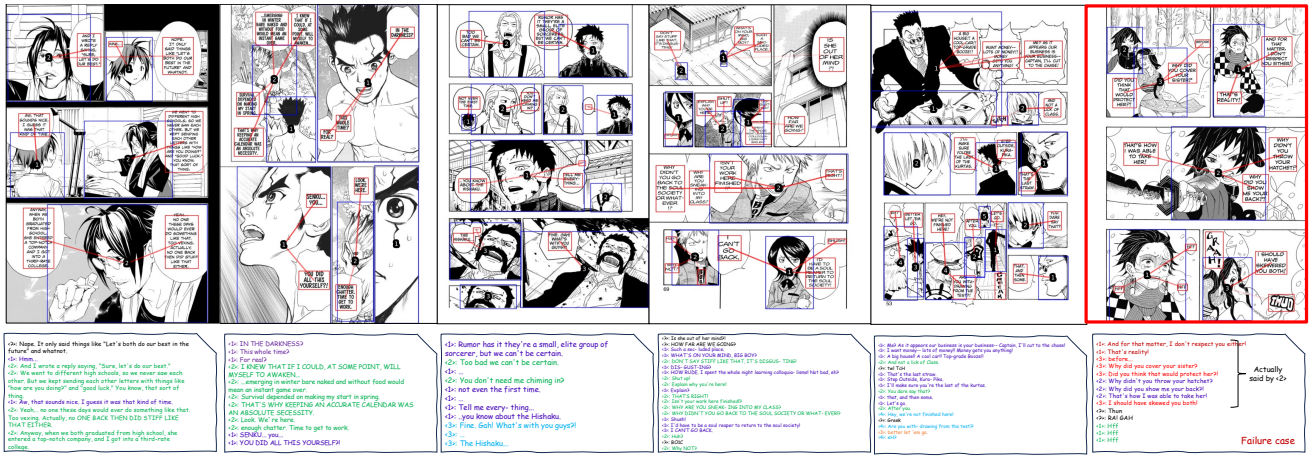

Figure 5. Text to speaker predictions generated by the *Magi* model. Each predicted text box is connected to a predicted character box using a line. The opacity of the line reflects the confidence of the model (the darker the line, the more confident the model is). Each predicted character box has a number at its centre based on the clustering predictions. We also show the final generated transcript. Note that all the dialogues are in the correct reading order. For text to speaker predictions that have a low confidence score (< 0.4) we replace the predicted speaker with $\langle ? \rangle$ in the generated transcript and let the reader infer it from context. Best viewed digitally.

Mangadex-1.5M: A large scale dataset of $\approx$ 1.5 million manga pages spanning multiple genres and art styles. A large majority of these are not made by professionals, and therefore, on average the images are "simpler" than PopManga. We use this dataset for pretraining.

5.2. Dataset curation and annotations

PopManga

Curation: 57000+ images are downloaded from various sources including Mangadex [27] and MangaPlus [28].

Annotation: We get human annotators to label text boxes, character boxes, character-character association and text-character association using a modified version of the LISA annotation tool [9]. For panels, we first finetune DETR [5] to predict panels using Manga109 and then use it to automatically generate pseudo panel annotations for the *Dev* subset of PopManga (we found these generated panels to be of satisfactory quality, thus reducing annotation cost). In the arXiv version of the paper we include more information on the annotation process.

Mangadex-1.5M

Curation: The $\approx$ 1.5 million *unlabelled* manga pages are downloaded from Mangadex [27] using their official API. During curation, we only query for manga which are available in English and have a safe-for-work content rating.
Annotation: For the collected images we do not have ground truth annotations. Instead, we generate and mine pseudo-annotations for them. *(1) Detection:* We prompt Grounding-DINO [24] (which has a strong zero-shot open-vocabulary detection performance) to detect bounding boxes for characters (using "human" and "character" as prompt) and text blocks (using "text" as prompt). We use a finetuned DETR to generated psuedo panel annotations, as done for PopManga *Dev*. *(2) Character-Character association*: We mine positive/negative character pairs for each page using the following strategy: (a) if character A and character B are in the same panel, they are not the same character (negatives); (b) if character A and character B are nearest neighbours of each other in the model's latent space, then A and B are the same characters (positives), as long as A and B are not in the same panel; (c) if $A = B$ and $B \neq C$, then $A \neq C$ (more negatives via transitivity). *(3) Text-Character association*: Here we simply match each text block to the closest character. This is not always correct, but it is correct more often than not.

6. Results

In the following we report results on the set of tasks that are needed to realise our goal of diarisation. For baselines we use state-of-the-art models that are available publicly.

6.1. Test sets and Baselines

Character detection
Datasets: PopManga (Test-S), PopManga (Test-U) and Manga109 (same test split as [44]).
Baselines: We compare our model against state-of-the-art character detection model DASS [44], and zero-shot results from Grounding-DINO [24].
Text detection
Datasets: PopManga (Test-S) and PopManga (Test-U). We do not report results on Manga109 as it does not contain English texts.
Baselines: We compare our model against the zero-shot results from Grounding-DINO [24].

Panel detection
Datasets: Manga109 (same test split as [44]).
Baselines: We compare our model against the zero-shot results from Grounding-DINO [24].

Character clustering
Datasets: PopManga (Test-S), PopManga (Test-U) and Manga109 (same test split as [45]).
Baselines: We compare our model against AMFR [45], which is a manga face clustering model, ZACI [2], which is a zero-shot anime face identification model. In addition, we use large pretrained vision encoders, CLIP [38] and DINO-V2 [34], to evaluate their feature representation for manga characters. Note that the baselines operate on cropped characters, unlike our method which operates directly on the entire manga page.

Speaker association
Datasets: PopManga (Test-S) and PopManga (Test-U). We do not report results on Manga109 as it does not contain English texts.
Baselines: To the best of our knowledge, there are no openly available models that can be adapted for our purposes. Instead we use "match each text box to the closest character" heuristic as a simple baseline.

6.2. Evaluation Metrics

Detection: We follow the object detection literature and evaluate our method using the average precision metric, as defined by [22]. For a predicted box to be positive, we use an IOU threshold of 0.5. Since methods being evaluated generate a different number of bounding boxes, we use the top 100 predictions only for a fair comparison.

Character clustering: We evaluate our method using several metrics that are commonly used in the metric learning literature. These metrics can be categorised into (1) retrieval based metrics—Mean Reciprocal Rank (MRR), Mean Average Precision at R (MAP@R), Precision at 1 (P@1), R-Precision (R-P)—that essentially operate on nearest neighbours of a query sample, (2) clustering based metrics—Adjusted Mutual Info (AMI), Normalised Mutual Info (NMI)—that evaluate the quality of predicted clusters and hard decision boundaries. We compute these metrics for each manga page and average over the entire test set.

Speaker association: We evaluate our method using Recall@#text metric, as formulated in [21]. This is similar to the Recall@K metric [26], except K is the number of texts per page. This metric is computed for each manga page and averaged over the entire test set.

	PopManga (Test-S)		PopManga (Test-U)		Manga109	
method	Char	Text	Char	Text	Body	Panel
DASS [44]	0.8410	-	0.8580	-	**0.9251**	-
Grounding-DINO [24]	0.7250	0.7922	0.7420	0.8301	0.7985	0.5131
Magi [Ours]	**0.8485**	**0.9227**	**0.8615**	**0.9208**	0.9015	**0.9357**

Table 2. Detection Results. We report the average precision results, which have an upper bound of 1.0.

method	AMI	NMI	MRR	MAP@R	P@1	R-P
	PopManga (Test-S)					
DINO-V2 [34]	0.0704	0.6219	0.6549	0.3870	0.4851	0.4390
CLIP [38]	0.1053	0.5405	0.7237	0.4757	0.5836	0.5189
ZACI [2]	0.0510	0.5656	0.6470	0.3806	0.4687	0.4354
AMFR [45]	0.1550	0.5355	0.7404	0.5004	0.6085	0.5397
Magi (crop only) [Ours]	0.4342	0.6734	0.8808	0.7452	0.8099	0.7655
Magi [Ours]	**0.6551**	**0.8505**	**0.9325**	**0.8448**	**0.8896**	**0.8564**
	PopManga (Test-U)					
DINO-V2 [34]	0.0806	0.6885	0.6281	0.3621	0.4420	0.4135
CLIP [38]	0.1057	0.5691	0.7145	0.4623	0.5636	0.5027
ZACI [2]	0.0767	0.3354	0.6409	0.3797	0.4603	0.4276
AMFR [45]	0.1828	0.6141	0.7570	0.5319	0.6235	0.5676
Magi (crop only) [Ours]	0.4387	0.7022	0.8801	0.7445	0.8059	0.7626
Magi [Ours]	**0.6509**	**0.8495**	**0.9331**	**0.8512**	**0.8905**	**0.8611**
	Manga109 (face)					
DINO-V2 [34]	0.1788	0.5795	0.7451	0.5108	0.5918	0.5468
CLIP [38]	0.1990	0.6502	0.7698	0.5466	0.6263	0.5798
ZACI [2]	0.2428	0.5231	0.7970	0.5951	0.6679	0.6245
AMFR [45]	0.2814	0.5739	0.8167	0.6272	0.7003	0.6536
Magi (crop only) [Ours]	0.3182	0.6078	0.8233	0.6370	0.7099	0.6628
Magi [Ours]	**0.5369**	**0.7843**	**0.9148**	**0.8151**	**0.8558**	**0.8279**
	Manga109 (body)					
DINO-V2 [34]	0.1077	0.5248	0.6963	0.4337	0.5183	0.4814
CLIP [38]	0.1643	0.5034	0.7542	0.5164	0.6027	0.5591
ZACI [2]	0.1090	0.4504	0.6919	0.4289	0.5116	0.4767
AMFR [45]	0.2622	0.6054	0.8076	0.5980	0.6845	0.6293
Magi (crop only) [Ours]	0.5132	0.7322	0.9052	0.7887	0.8418	0.8050
Magi [Ours]	**0.6345**	**0.8202**	**0.9383**	**0.8567**	**0.8966**	**0.8667**

Table 3. Character Clustering Results. We report results using several metrics. They all have an upper bound of 1.0.

method	PopManga (Test-S)	PopManga (Test-U)
shortest distance	0.7758	0.7659
Magi [Ours]	**0.8448**	**0.8313**

Table 4. Speaker Association Results. We report the Recall@#text results, which have an upper bound of 1.0.

6.3. Performance and Discussion

We report quantitative results in Tables 2, 3, and 4 and show qualitative results in Figures 4 and 5. These results demonstrate the efficacy of our model and its impressive performance over the baselines.

Detection: For the task of detecting texts and panels, our method performs markedly better compared to existing solutions. For characters, our model's performance is comparable to DASS [44], surpassing it in 2 out of 3 test sets, with a slight concession in the remaining third. Notably, Grounding-DINO [24] shows a strong zero-shot detection performance for characters and texts. However, its effectiveness takes a discernible dip when applied to panels.

Character clustering: The biggest strength of our model, *Magi*, lies in its ability to perform "in-context" character clustering, establishing itself as a state-of-the-art solution across all metrics and test sets. Particularly the metric of interest is AMI, which evaluates clustering performance and the model's ability to form decisive boundaries. Note that AMI is adjusted such that a random clustering has a score of 0 whereas in NMI random clusters may still get a positive score. Our model's AMI results significantly outshine those of existing methods, showcasing superiority by several magnitudes. Additionally, the results of our crop-only embedding module, though superior to baselines, fall short of the performance achieved by our holistic model, *Magi*, with context. This underscores our hypothesis that operating only on crops loses visual cues on the page that can assist in clustering.

In Figure 4, our model demonstrates its prowess in character clustering, successfully handling scenarios where characters are observed from disparate viewpoints, with partial visibility, at various scales, and diverse expressions. Notably, a failure case is presented where the model erroneously matched two characters with similar checked clothing, emphasising the challenges posed by subtle visual details, even when the characters are distinct (in this case, brothers). Nevertheless, it is evident that the model is able to pick up on multiple cues for character matching, including clothing, hair style, and the faces.

Speaker Association: In Figure 5, we show that our method can adeptly match texts to their respective speakers. We also demonstrate the use of the confidence values to filter cases where the model is unsure, which includes non-dialogue texts (e.g. sound effects) or texts with no character in the panel. The task of matching texts to their speakers in manga is challenging, often necessitating an understanding of conversation history and context to disambiguate speakers.

7. Conclusion

In this study, our primary objective was to improve the accessibility of manga for individuals with visual impairments. Tackling the complex task of diarisation, we have laid the groundwork for a fully automated transcription of manga content, enabling active engagement for everyone, irrespective of their visual abilities.

Looking ahead, numerous promising avenues for future research beckon. Particularly, we anticipate leveraging the language understanding capabilities of Large Language Models (LLMs) to enhance diarisation by incorporating conversation history and context.

Acknowledgements: We would like to thank Cindy Seuk and Anhad Sachdeva for their assistance with annotation quality assurance. This research is supported by EPSRC Programme Grant VisualAI EP/T028572/1 and a Royal Society Research Professorship RP\R1\191132.

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. 2, 5

[2] Kosuke Akimoto. Danbooru 2020 zero-shot anime character identification dataset (zaci-20). https://github.com/kosuke1701/ZACI-20-dataset, 2021. 7, 8

[3] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. Coo: Comic onomatopoeia dataset for recognizing arbitrary or truncated texts. In *European Conference on Computer Vision*, pages 267–283. Springer, 2022. 2

[4] Maciej Budyś. Manga ocr. https://github.com/kha-white/manga-ocr, 2022. 5

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3, 7

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2

[7] Des livres à voir et à toucher. Des livres à voir et à toucher. https://www.lavillebraille.fr/des-livres-a-voir-et-a-toucher/. 2

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[9] Abhishek Dutta and Andrew Zisserman. The via annotation software for images, audio and video. In *ACM MM*, New York, USA, 2019. ACM, ACM. to appear in Proceedings of the 27th ACM International Conference on Multimedia (MM 19). 7

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Doll'ar, and Ross B Girshick. Masked autoencoders are scalable vision learners. 2022 ieee. In *CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15979–15988, 2021. 4

[12] Zheqi He, Yafeng Zhou, Yongtao Wang, Siwei Wang, Xiaoqing Lu, Zhi Tang, and Ling Cai. An end-to-end quadrilateral regression network for comic panel extraction. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 887–895, 2018. 2

[13] Ryota Hinami, Shonosuke Ishiwatari, Kazuhiko Yasuda, and Yusuke Matsui. Towards fully automated manga translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12998–13008, 2021. 5

[14] Naoto Inoue, Ryosuke Furuta, Toshihiko Yamasaki, and Kiyoharu Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5001–5009, 2018. 2

[15] Junguang Jiang, Baixu Chen, Jianmin Wang, and Mingsheng Long. Decoupled adaptation for cross-domain object detection. In *International Conference on Learning Representations*, 2022. 2

[16] Arthur B Kahn. Topological sorting of large networks. *Communications of the ACM*, 5(11):558–562, 1962. 5

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 4

[18] Samu Kovanen and Kiyoharu Aizawa. A layered method for determining manga text bubble reading order. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4283–4287, 2015. 5

[19] Yunjung Lee, Hwayeon Joh, Suhyeon Yoo, and Uran Oh. Accesscomics: an accessible digital comic book reader for people with visual impairments. In *Proceedings of the 18th International Web for All Conference*, pages 1–11, 2021. 2

[20] Minghao Li, Tengchao Lv, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. Trocr: transformer-based optical character recognition with pre-trained models. arxiv 2021. *arXiv preprint arXiv:2109.10282*, 2021. 5

[21] Yingxuan Li, Kiyoharu Aizawa, and Yusuke Matsui. Manga109dialog a large-scale dialogue dataset for comics speaker detection. *arXiv preprint arXiv:2306.17469*, 2023. 2, 7

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 7

[23] Little Prince in braille! Little prince in braille! https://www.blog.thelittleprince.com/little-prince-in-braille/. 2

[24] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 7, 8

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 4

[26] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 852–869. Springer, 2016. 7

[27] MangaDex. Mangadex. https://mangadex.org/. 6, 7

[28] MangaPlus. Mangaplus by shueisha. https://mangaplus.shueisha.co.jp/. 5, 6

[29] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3651–3660, 2021. 3, 4

[30] Philipp Meyer. Life - a tactical comic for the blind people. 2013. 2

[31] MyAnimeList. Top manga - myanimelist.net. https://myanimelist.net/topmanga.php. 5

[32] Nhu-Van Nguyen, Christophe Rigaud, and Jean-Christophe Burie. Comic mtl: optimized multi-task learning for comic book image analysis. *International Journal on Document Analysis and Recognition (IJDAR)*, 22:265–284, 2019. 2

[33] Toru Ogawa, Atsushi Otsubo, Rei Narita, Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Object detection for comics using manga109 annotations. *arXiv preprint arXiv:1803.08670*, 2018. 2

[34] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 7, 8

[35] Xufang Pang, Ying Cao, Rynson WH Lau, and Antoni B Chan. A robust panel extraction method for manga. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 1125–1128, 2014. 2

[36] Boonyarith Piriyothinkul, Kitsuchart Pasupa, and Masanori Sugimoto. Detecting text in manga using stroke width transform. In *2019 11th International Conference on Knowledge and Smart Technology (KST)*, pages 142–147. IEEE, 2019. 2

[37] Xiaoran Qin, Yafeng Zhou, Yonggang Li, Siwei Wang, Yongtao Wang, and Zhi Tang. Progressive deep feature learning for manga character recognition via unlabeled training data. In *Proceedings of the ACM Turing Celebration Conference-China*, pages 1–6, 2019. 2, 3

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 7, 8

[39] Reshma Ramaprasad. Comics for everyone: Generating accessible text descriptions for comic strips. *arXiv preprint arXiv:2310.00698*, 2023. 2

[40] Christophe Rigaud, Nam Le Thanh, J-C Burie, J-M Ogier, Motoi Iwata, Eiki Imazu, and Koichi Kise. Speech balloon and speaker association for comics and manga understanding. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 351–355. IEEE, 2015. 2

[41] Suprosanna Shit, Rajat Koner, Bastian Wittmann, Johannes Paetzold, Ivan Ezhov, Hongwei Li, Jiazhen Pan, Sahand Sharifzadeh, Georgios Kaissis, Volker Tresp, et al. Relationformer: A unified framework for image-to-graph generation. In *European Conference on Computer Vision*, pages 422–439. Springer, 2022. 3

[42] Gürkan Soykan, Deniz Yuret, and Tevfik Metin Sezgin. Identity-aware semi-supervised learning for comic character re-identification. *arXiv preprint arXiv:2308.09096*, 2023. 2, 3

[43] Star Wars. Star wars audio comics. https://www.youtube.com/@StarWarsAudioComics/. 2

[44] Barış Batuhan Topal, Deniz Yuret, and Tevfik Metin Sezgin. Domain-adaptive self-supervised pre-training for face & body detection in drawings. *arXiv preprint arXiv:2211.10641*, 2022. 2, 7, 8

[45] Koki Tsubota, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Adaptation of manga face representation for accurate clustering. In *SIGGRAPH Asia 2018 Posters*, pages 1–2. 2018. 2, 3, 7, 8

[46] Yongtao Wang, Yafeng Zhou, and Zhi Tang. Comic frame extraction via line segments combination. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 856–860. IEEE, 2015. 2

[47] Zhimin Zhang, Zheng Wang, and Wei Hu. Unsupervised manga character re-identification via face-body and spatial-temporal associated clustering. *arXiv preprint arXiv:2204.04621*, 2022. 2, 3

[48] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 3

[49] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3