Towards Scene Text Translation for Complex Writing Systems

Hour Kaing[1] Haiyue Song[1] Chenchen Ding[1] Jiannan Mao[2] Hideki Tanaka[1] Masao Utiyama[1]
[1]ASTREC, UCRI, NICT, Japan [2]Gifu University, Gifu, Japan
[1]{hour_kaing, haiyue.song, chenchen.ding, hideki.tanaka, mutiyama}@nict.go.jp
[2]mao@mat.info.gifu-u.ac.jp

Abstract

Scene text translation aims to automatically translate text in images or videos while preserving its visual features. In this work, we focus on scene text translation for complex writing system by taking Japanese as a typical example. We build a pipeline to translate from English to Japanese, leveraging publicly available modules for text detection, recognition, and translation, and train our own text replacement model specialized for English-to-Japanese transformations. Experiments show that the system can effectively generate translated text in Japanese while retaining much of the original style, although background regeneration and handling of Kanji remain open challenges.

1 Introduction

When watching a foreign movie, untranslated on-screen text in the background often hinders understanding of the scene, as shown in Figure 1. Translating such text and placing it in the correct position with similar style typically requires significant manual effort.

Scene text translation systems [1] (also referred to as cross-language text editing systems [2]) provide an automatic solution by translating the source text in video scenes into the target language while preserving the visual features of the original text, such as its location, font, and background. This is typically achieved by integrating scene text detection and recognition, machine translation, and scene text replacement modules.

However, translating scene text into complex writing systems is challenging. Japanese can be regarded as a typical example of a complex writing system, which encompasses thousands of distinct characters across multiple forms (e.g., Kanji, Hiragana, Katakana).

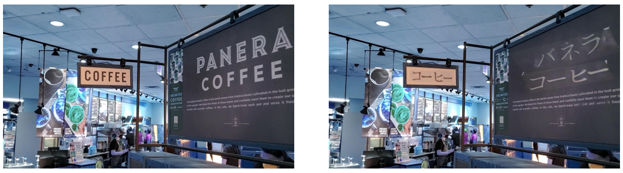

Figure 1 An example of translating scene text from English to Japanese in videos, while preserving the original position and style.

To address this, we train a specialized Japanese text replacement module by 1) synthesizing 100k cross-lingual text images from English to Japanese, and 2) fine-tuning the English SRNet model [2] on our synthetic dataset. For other modules in our English-to-Japanese scene text translation system, we leverage publicly available models including the FAST text detection model [3], the CRNN text recognition model [4], and the NLLB200 machine translation model [5].

Experiments show that our model performs better than baselines in generating Japanese text but underperforms them in background regeneration. We also found that generating Kanji characters is more challenging than generating Katakana or Hiragana characters.

2 Background

We introduce the modules involved in our scene text translation system, including detecting and recognizing the English text in each frame (Section 2.1), translating it into the target language (Section 2.2), and placing it back into the original position with similar visual features (Section 2.3). Previous scene text translation systems are introduced in Section 2.4.

2.1 Text Detection and Recognition

Scene text detection aims to identify text regions in natural (often noisy) scenes [3, 6, 7, 8]. We adopt the FAST (faster arbitrarily-shaped text detector) [3] system, which achieves real-time, high-accuracy detection for curved text and supports multiple languages including Japanese.

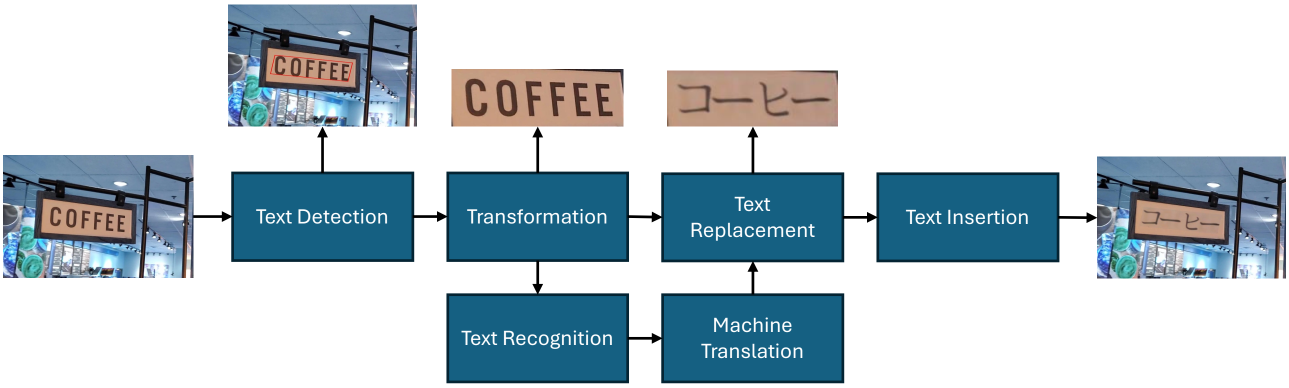

Figure 2 Scene Text Translation Pipeline.

Scene text recognition then extracts text from the detected sub-images. This is typically done with optical character recognition (OCR) models such as CRNN [4], SSDAN [9], or PaddleOCR.[1] In our work, we use the CRNN [4] system because it handles sequences without explicit character segmentation and supports non-Latin scripts such as Japanese or Chinese.

2.2 Machine Translation

Next, the extracted source text is translated into the target language through a separate neural machine translation (NMT) system [5, 10, 11]. We employ the NLLB200 [5] model, a state-of-the-art multilingual NMT system that provides high-quality English to Japanese translation.

Although multimodal machine translation (MMT) that uses images [12, 13] or videos [14, 15] as assisting information can yield better performance, we adopt an efficient text-to-text MT approach for real-time video translation.

2.3 Scene Text Replacement

Scene text replacement aims to edit the text in an image by replacing it with new text while preserving the original text style and background information. The model typically takes two inputs: a scene text image to be edited and the new text to be inserted. It then outputs a new image containing the new text. Previous works include SRNet [2], STEFANN [16], SwapText [17], and STRIVE [18]. We apply the SRNet architecture, which comprise a text conversion module, a background inpainting module, and a fusion module, since its code is publicly available.[2]

2.4 Scene Text Translation

Scene text translation has been explored in both research [1, 19, 20] and commercial applications (e.g., Google Translate's Camera mode). However, previous systems [19, 20] typically focus on image translation without adapting text style, or are limited to Indic scripts such as English to Hindi [1].

Our paper presents the first study to explore English to Japanese translation in video scenes, addressing the challenge of the large number of characters and complex scripts in Japanese.

3 Method

Our pipeline consists of six key modules—text detection, transformation, text recognition, machine translation, text replacement, and text insertion— as shown in Figure 2. The process begins with an input image containing text, from which the pipeline detects text regions, crops them, and transforms these regions into a rectangular shape. The transformed text image is then recognized and translated from English to Japanese. Next, given the transformed text image and the translated Japanese text, the text replacement module generates a new text image in Japanese with the same background and a visually similar text style. Finally, the translated text image is inserted back into the original image, replacing the corresponding English text region.

In this work, we employed publicly available models wherever possible. Specifically, we used the FAST model [3] for text detection, the CRNN model [4] for text recognition, and the NLLB200 model [5] for machine translation. For the transformation module, we adopted perspective transformation, a geometric function that reshapes a quadrilateral image into a rectangular one. Con-

[1] https://github.com/PaddlePaddle/PaddleOCR
[2] https://github.com/lksshw/SRNet

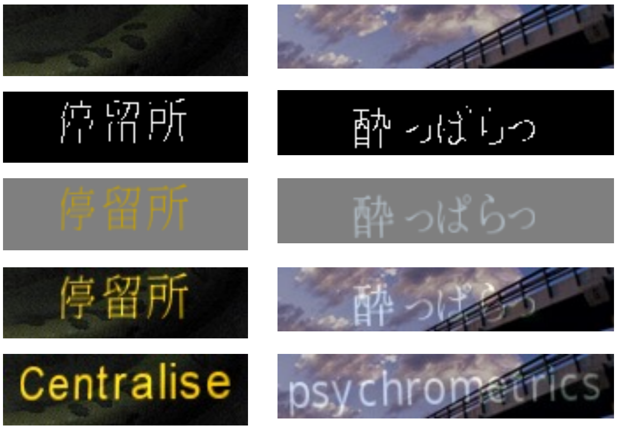

Figure 3 Examples of synthetic data. From top to bottom: background, text skeleton, foreground text, target image with Japanese text, source image with English text.

versely, for the insertion module, we utilized inverse perspective transformation to accurately position the translated text image within the original parent image.

For text replacement, we trained our own model because no pre-existing cross-lingual text replacement model (from English to Japanese) was available. Specifically, we synthesized 100k sets of images as the training data of our text replacement model. Since SRNet is trained on multiple objectives to optimize text conversion, background inpainting, and the fusion module, we synthesize the following for each set: two style images (one for English and one for Japanese),[3] a background image, a foreground Japanese text image, and a Japanese text skeleton image, as shown in Figure 3. The style of text is generated by randomly selecting text fonts, color, and parameters of deformation, and background image is randomly chosen from a background set. This data synthetic process follows the design of the English text replacement approach described in [2]. The primary difference is that our target texts are in Japanese, randomly selected from a set of 45k Japanese words.[4] Additionally, we leveraged an existing English model by fine-tuning SRNet$_{en}$ [2] on our synthetic dataset, where the input is English style image and the output is the paired Japanese style image, resulting in xSRNet$_{enja}$. Since the original SRNet$_{en}$ weights were not publicly released, we fine-tuned a reproduced version of the model instead. Both the synthetic tool and the SRNet$_{en}$ model are available in a public repository.[5]

Model	MSE↓	PSNR↑	SSIM↑
EnSource$_{in}$	29.79	33.65	0.61
Background$_{ref}$	**18.71**	**35.91**	**0.75**
Foreground$_{ref}$	95.65	28.60	0.46
SRNet$_{en}$	54.55	31.01	0.57
xSRNet$_{enja}$ (ours)	73.94	29.66	0.63

Table 1 Quality assessment of cross-lingual text replacement. Bold and underlined scores are first and second best, respectively.

4 Evaluation

In this section, we assess our pipeline by first examining the quality of text replacement and then evaluating the overall performance of our scene text translation pipeline.

4.1 Quality of Text Replacement

We measured the quality of text replacement using Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity (SSIM) metrics [21]. To compute these metrics, the evaluation dataset must comprise of English text images as model inputs and Japanese text images as references, of which English and Japanese texts are mutually translated. Due to the absence of such dataset for the English-Japanese pairs, we synthesized 1k images using a distinct set of background images[6] and pairwise English-Japanese text data,[7] all of which differ from our synthetic training dataset. Table 1 summarizes the performance of our xSRNet$_{enja}$ model, evaluated via MSE, PSNR, and SSIM. We compare our model against SRNet$_{en}$, which was trained solely for English text replacement and applied here for cross-lingual text replacement. We further include three strong baselines that share part of reference by computing evaluation metrics on ground-truth Japanese text images against their respective: (1) source image with English text (EnSource$_{in}$), (2) ground-truth background images without text on it (Background$_{ref}$), and (3) forground Japanese text with a grey background (Foreground$_{ref}$). The samples of (1), (2), and (3) are similar to those in Figure 3 at line five, one, and three, respectively.

As a result, Background$_{ref}$ achieves the best performance across all metrics, which is not surprising given that the background in the evaluation dataset is challenging, and neither SRNet$_{en}$ nor xSRNet$_{enja}$ is expected to perfectly regenerate the background. Our model, xSRNet$_{enja}$, un-

[3] They are not necessary to be parallel words.
[4] https://github.com/hingston/japanese
[5] https://github.com/lksshw/SRNet
[6] https://github.com/clovaai/synthtiger
[7] https://github.com/facebookresearch/MUSE

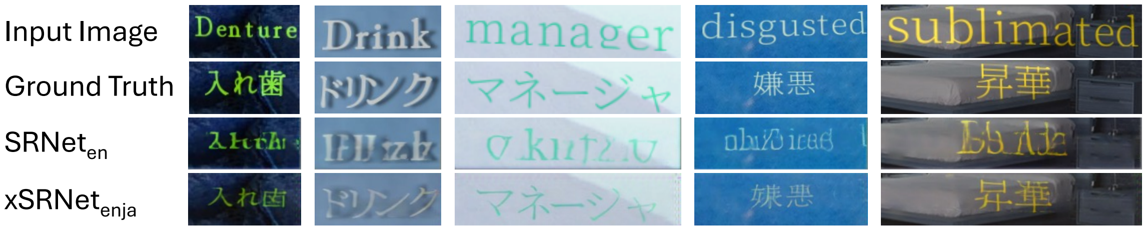

Figure 4 Samples generated by text replacement modules.

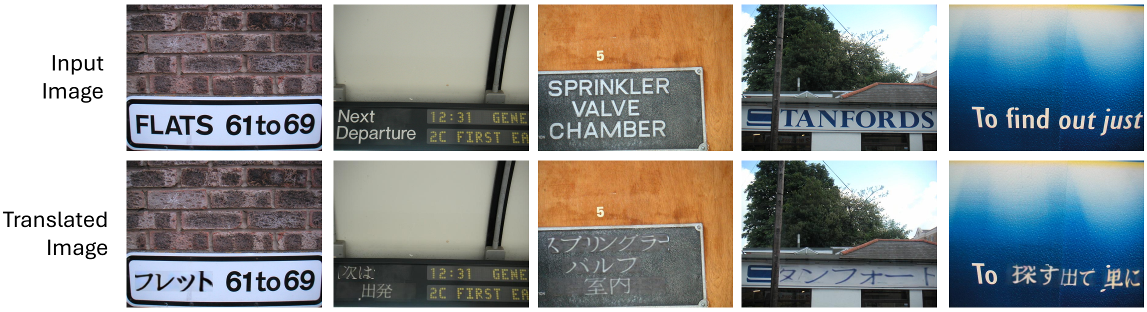

Figure 5 Samples generated by our scene text translation pipeline.

derperforms compared to enSceneImg_{in} and even SRNet_{en} in terms of MSE and PSNR, indicating that further improvements in background regeneration are needed. However, instead of focusing solely on absolute errors measured by MSE and PSNR, our model outperforms SRNet_{en} when measured by structural similarity (SSIM). This suggests that our model tends to generate Japanese text images that are more structurally similar to the ground truth, as illustrated in Figure 4. Additionally, we observe that generating Kanji characters is more challenging than generating Katakana characters.

4.2 Quality of Scene Text Translation

To this end, we assessed the quality of our pipeline when all modules were used together. We applied our approach to the ICDAR 2003 scene text dataset [22]. Figure 5 illustrates several selected samples that were translated using our pipeline. These samples are relatively straightforward for our text replacement module because the detected text regions include fewer noise elements. However, errors still occur due to other modules. For instance, some text remained in English because the text detection module failed to detect it; other text was mistranslated owing to stylistic features of the original text image, such as “STANFORDS” with a mixed font style between the letter “S” and the rest. All these observations suggest that further improvements are needed not only in the text replacement module but also in other components, including text detection, recognition, and translation.

5 Conclusion and Future Works

We have presented a pipeline to translate scene text images from English to Japanese, utilizing open models, except for the text replacement model, which was trained by ourselves. We demonstrated that our pipeline is capable of translating English scene text images, though it has some limitations, such as difficulty in generating Kanji characters, inability to detect all text, and restricting detection and translation to words rather than phrases or sentences.

There is plenty of room to improve our pipeline. First, we aim to enhance the text replacement module by using more synthesized training images with diverse background scenes and Japanese Kanji characters. For text detection, we will consider combining multiple text detection models to ensure that all texts are translated. Furthermore, since scene text is not always limited to individual words, we will also explore contextual translation of phrases or sentences.

Finally, although our pipeline can be used to translate text in videos frame-by-frame, the translation in the resulting video may appear inconsistent. This occurs because certain frames may be blurry or contain non-frontal text, posing particular challenges for text detection. Additional modules, such as reference frame selection and text propagation [18], are needed to achieve more consistent and fluid translations in video. Addressing these challenges is part of our future work.

References

[1]Shreyas Vaidya, Arvind Kumar Sharma, Prajwal Gatti, and Anand Mishra. Show me the world in my language: Establishing the first baseline for scene-text to scene-text translation. In **ICPR**, 2024.

[2]Liang Wu, Chengquan Zhang, Jiaming Liu, Junyu Han, Jingtuo Liu, Errui Ding, and Xiang Bai. Editing text in the wild. In **Proc ACM Int Conf Multimed**, pp. 1500–1508, 2019.

[3]Zhe Chen, Jiahao Wang, Wenhai Wang, Guo Chen, Enze Xie, Ping Luo, and Tong Lu. Fast: Faster arbitrarily-shaped text detector with minimalist kernel representation. **arXiv:2111.02394**, 2021.

[4]Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. **IEEE PAMI**, Vol. 39, No. 11, pp. 2298–2304, 2016.

[5]NLLB Teamg. No language left behind: Scaling human-centered machine translation, 2022.

[6]Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. EAST: An efficient and accurate scene text detector. In **CVPR**, pp. 5551–5560, 2017.

[7]Wenhai Wang, Enze Xie, Xiang Li, Wenbo Hou, Tong Lu, Gang Yu, and Shuai Shao. Shape robust text detection with progressive scale expansion network. In **CVPR**, pp. 9336–9345, 2019.

[8]Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In **AAAI**, Vol. 34, pp. 11474–11481, 2020.

[9]Yaping Zhang, Shuai Nie, Wenju Liu, Xing Xu, Dongxiang Zhang, and Heng Tao Shen. Sequence-to-sequence domain adaptation network for robust text image recognition. In **CVPR**, pp. 2735–2744, 2019.

[10]Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

[11]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[12]Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. Multi30K: Multilingual English-German image descriptions. In **VL**, pp. 70–74, 2016.

[13]Lucia Specia, Stella Frank, Khalil Sima'an, and Desmond Elliott. A shared task on multimodal machine translation and crosslingual image description. In **WMT**, pp. 543–553, 2016.

[14]Tosho Hirasawa, Zhishen Yang, Mamoru Komachi, and Naoaki Okazaki. Keyframe segmentation and positional encoding for video-guided machine translation challenge 2020, 2020.

[15]Weiqi Gu, Haiyue Song, Chenhui Chu, and Sadao Kurohashi. Video-guided machine translation with spatial hierarchical attention network. In **ACL—IJCNLP**, pp. 87–92, 2021.

[16]Prasun Roy, Saumik Bhattacharya, Subhankar Ghosh, and Umapada Pal. Stefann: Scene text editor using font adaptive neural network. In **CVPR**, June 2020.

[17]Qiangpeng Yang, Hongsheng Jin, Jun Huang, and Wei Lin. Swaptext: Image based texts transfer in scenes, 2020.

[18]Vijay Kumar BG, Jeyasri Subramanian, Varnith Chordia, Eugene Bart, Shaobo Fang, Kelly Guan, and Raja Bala. Strive: scene text replacement in videos. In **ICCV**, pp. 14529–14538. IEEE, 2021.

[19]Puneet Jain, Orhan Firat, Qi Ge, and Sihang Liang. Image translation network. 2021.

[20]Zhibin Lan, Jiawei Yu, Xiang Li, Wen Zhang, Jian Luan, Bin Wang, Degen Huang, and Jinsong Su. Exploring better text image translation with multimodal codebook. In **ACL**, pp. 3479–3491, 2023.

[21]Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. **IEEE transactions on image processing**, Vol. 13, No. 4, pp. 600–612, 2004.

[22]Simon M Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, Robert Young, Kazuki Ashida, Hiroki Nagai, Masayuki Okamoto, Hiroaki Yamamoto, et al. Icdar 2003 robust reading competitions: entries, results, and future directions. **IJDAR**, Vol. 7, pp. 105–122, 2005.