부산대학교

정보컴퓨터공학부

자연스러운 번역 텍스트 합성 착수보고서

이동훈, 이승재, 문경환

Supervisor: 전상률

July 16, 2025

Contents

Chapter 1

Inroduction

1.1 Research Purpose

본 연구는 기존의 기계 번역 및 OCR 기반 텍스트 삽입 방식이 갖는 한계를 분석하고, 보다 자연스러운 번역 텍스트 합성을 실현하기 위한 기술적 요구 사항과 구현 방안을 탐구한다. 나아가, 사용자가 입력한 이미지에 적절하게 번역 텍스트를 자연스럽게 삽입할 수 있는 애플리케이션 서비스를 제안하고 구현하는 것을 궁극적인 목표로 한다.

1.2 자문 개요

본 자문은 현대자동차 책임연구원 조재훈 박사가 참여하였다. 내용은 아래와 같다.

중간 보고서 보완 제안

1. **기술 구현의 세부 사항 추가**
   - "자연스러움"의 기준은 주관적이므로 SSIM, PSNR 등의 지표나 사용자 테스트를 통해 정량적 평가 체계를 마련할 필요가 있음.
   - PaddleOCR의 구성(예: detection model, recognition model, 전처리 과정 등)을 구체적으로 명시할 것.
   - GaRNet, TextCtrl의 주요 하이퍼파라미터 설정 및 선택 근거(예: ROI 방식, attention 조절 등)를 기술할 것.
2. **기술적 한계 및 해결 방안 고도화**
   - 곡면/곡률 이미지 대응을 위해 depth estimation, normal map 보정 등 3D 변형 처리 기술을 검토할 필요 있음.
   - 연산 비용을 줄이기 위해 PP-OCR Tiny, TextCtrl-Lite 등 경량 모델의 적용 가능성을 고려해야 함.
3. **테스트 및 검증 시나리오 강화**
   - 예시 이미지를 통한 사용자 평가(MOS 방식)를 설계하고, 자연스러움 및 가독성에 대한 정성적 · 정량적 평가를 수행할 것.
   - 다국어 폰트 및 다양한 타이포그래피 실험을 통해 모델의 범용성을 검증할 것.

결론

본 보고서는 시각 언어 합성 문제를 기술적으로 잘 분석하고 있으며, 단계별 프레임워크 구성 또한 타당하다. 최종 결과물에서는 각 모듈의 성능 지표를 보완하고, 예시 이미지를 통해 기술 성과를 시각적으로 제시하는 것이 중요하다.

Chapter 2

Modifications According to Requirements

2.1 기술 구현의 세부 사항 추가

- 본 모델의 기존 평가지표가 불명확 하였기에, 기존 연구에서 널리 사용되는 PSNR과 User study를 추가한다.
- TextCtrl의 주요 하이퍼파라미터 설정은 기존 논문에서 구현된 파라미터를 따르며, 이는 사전에 기존 논문에서 최적의 파라미터를 구성하였기에 선택하였다. 한 편 TextCtrl 모델에서 기존 배경을 기반으로 이미지를 생성하기에, GarNet은 사용하지 않는다 Zeng et al. (2024).
- PaddleOCR의 구조는 아래 섹션 2.1에서 기술한다.

PaddleOCR

PaddleOCR는 입력 이미지로부터 텍스트를 검출하고 인식하는 End-to-End OCR 파이프라인으로, 다음과 같은 구성요소로 구성된다 Cui et al. (2025).

1. 전처리 (Preprocessing)

- 입력 이미지 크기 조정 (비율 유지, 패딩 포함)
- 픽셀 정규화 및 데이터 타입 변환 (float32)
- 채널 순서 변경 (BGR $\rightarrow$ RGB)
- 입력 텐서로 변환 (CHW 형식)

2. 텍스트 검출 모델 (Text Detection)

이미지 내에서 텍스트가 존재하는 영역을 검출한다. 대표적인 모델은 다음과 같다:

- **DB (Differentiable Binarization)**: 정확하고 빠른 검출 성능 제공
- **EAST**: 초기 OCR 분야에서 널리 사용된 검출기
- **SAST**: 방향 인식이 가능하며 회전된 텍스트에 강건
- **PSE (Progressive Scale Expansion)**: 텍스트 인스턴스 클러스터링 기반

출력은 각 텍스트 영역의 다각형 좌표들이다.

3. 텍스트 인식 모델 (Text Recognition)

검출된 각 텍스트 영역을 자른 후 해당 영역 내의 문자를 인식한다. 주요 인식 모델은 다음과 같다:

- **CRNN**: CNN + BiLSTM + CTC 구조의 고전적인 인식기
- **SVTR**: Vision Transformer 기반 인식기로 고정도 성능 제공
- **RARE**: Spatial Transformer를 통해 왜곡된 문자를 보정하여 인식
- **Rosetta**: Facebook에서 제안한 경량 OCR 모델

4. 후처리 (Postprocessing)

- CTC decoding
- Direction classifier를 통한 좌우 반전 여부 판단
- Character dictionary 기반 문자 필터링
- Confidence score를 활용한 결과 필터링

2.2 기술적 한계 및 해결 방안 고도화

- 곡면/곡률 이미지 대응은 현재 모든 구현을 완료하고 보완이 필요하여 Future study로 지정한다.
- 연산 비용을 줄이기 위해 PP-OCR Tiny, TextCtrl-Lite 등 경량 모델의 적용 가능성을 제안하였으나, 리서치 결과 해당 모델들은 사용할 수 없거나 존재하지 않음을 확인하였다. 한 편 TextCtrl에서 텍스트를 지우는 프로세스가 내포되기에 GarNet을 사용하지 않음에 따른 전체 파이프라인 경량화가 진행되었다.

2.3 테스트 및 검증 시나리오 강화

- 예시 이미지를 통한 사용자 평가를 계획해 구현 이후 구글 폼을 이용한 사용자 평가 수행 예정이다.
- 다국어 폰트(한글, 영어, 일본어)를 적용하기 위해 TextCtrl 모델을 재학습한다. 본 보고서에서 뉴스기사 크롤링을 통한 6만자의 한글 데이터를 수집하였으며, 이를 통해 TextCtrl 모델을 재학습 하였다. TextCtrl 재학습 데이터는 입력한 배경 이미지 및, 그레이 스케일 배경과 폰트에 따라 생성된 한글 이미지를 합친 데이터이다. 이를 통해 PSNR을 기반으로 모델을 추가 언어에 대해 학습한다.

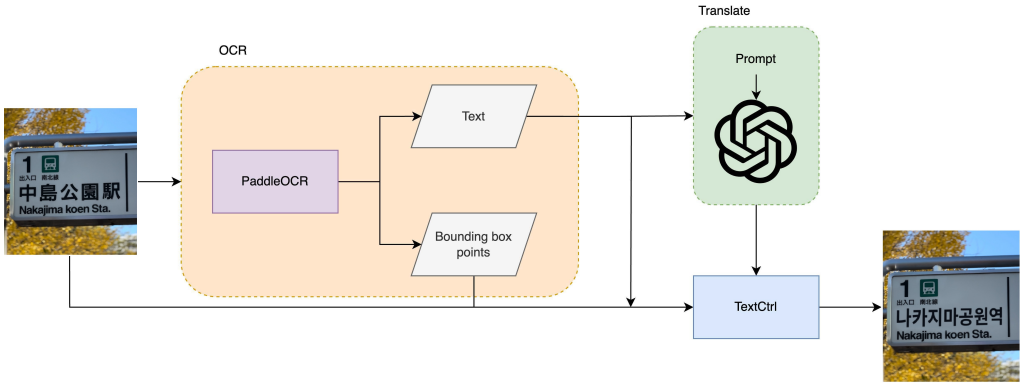


Figure 2.1: Overall architecture of proposed pipeline.

2.4 Details of design and changes

본 모델의 전체 구조는 그림 2.1와 같다. 프레임워크는 총 3단계로 구성되며, 기존의 텍스트 추출, 번역, 삭제 및 합성 단계에서 텍스트 추출, 번역, 합성 단계로 축소한다. 텍스트 추출 단계에서는 PaddleOCR 모델을 사용해 텍스트의 내용과 바운딩 박스를 추출한다. 이후 번역 단계에서 해당 텍스트를 번역하고, 원본 텍스트, 번역된 텍스트, 바운딩 박스 정보가 TextCtrl에 입력되어 최종 출력을 생성한다. Key modification과 수정 사항은 다음과 같다.

- 본 모델은 TextCtrl 모델의 이점을 사용해 GarNet의 이미지 삭제 단계를 제거하며 모델의 코스트를 완화아였다.
- 본 모델의 Pipeline 다양한 폰트와 추가 언어(한글)을 추가 학습하여, 모델의 언어 지원 다양성을 추가하였다.
- 모델의 추가 보완 가능성과 평가 지표를 추가 탐색하였다.

2.5 Progress

2.5.1 Time Table

타임 테이블은 그림 2.2과 같다.

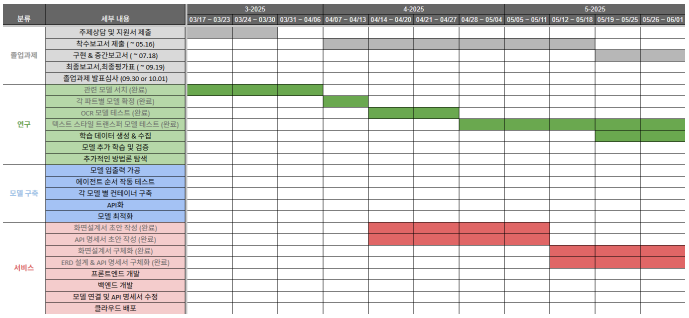
2.5.2 Personalized Progress

이동훈

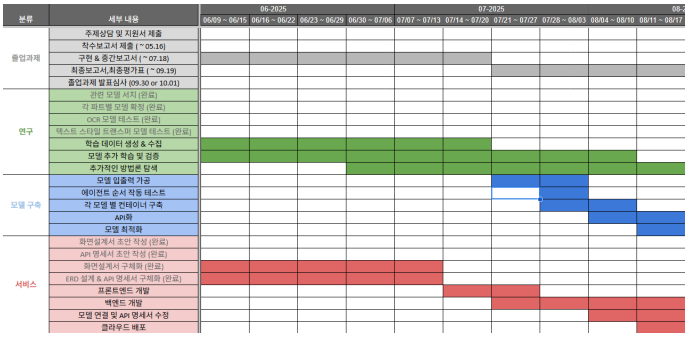
- 프로젝트 UI/UX 구체화
- 학습용 데이터셋 서치
- 학습용 폰트 수집

이승재

- SRnet gen-ko 데이터셋 생성
- style & glyph encoder 학습
- ko-to-ko textctrl ocr_loss 수정 후 본 학습에 반영 예정



(a)



(b)

(c)

Figure 2.2: The progress time table.

문경환

- 모델 평가지표 탐색 및 정리
- 보고서 및 문서 작성
- 학습용 한글 데이터 크롤링

2.6 Result

2.6.1 Textctrl: Dataset

기존 TextCtrl은 영문 텍스트 기반으로만 학습되어 있어, 한글 텍스트에 직접 적용하기 어려운 한계가 있었다. 이에 따라 본 연구에서는 데이터셋 생성을 위한 전처리 단계부터 새롭게 접근하였다.

SRNet-Datagen 기반의 기존 데이터 생성 코드를 한국어 환경에 맞게 수정하고, 한글 폰트가 바운딩 박스 안에 안정적으로 들어가도록 레이아웃을 조정했다. 특히 한글은 글자 구조상 자간 및 글자 크기의 변동 폭이 크기 때문에, 영문 대비 더 넓은 바운딩 박스를 적용하는 방식으로 데이터 생성을 진행했다. 이렇게 하여 총 20만 장 규모의 한글 텍스트 렌더링 이미지 데이터를 생성하였으며, 예시는 그림 2.3과 같다.

Figure 2.3: 왼쪽 위부터 오른쪽 아래 순서로 각각 `i_s` (배경 위 스타일 텍스트 a), `i_t` (회색 배경 위 일반 텍스트 b), `t_sk` (스타일 텍스트 b의 스켈레톤), `t_t` (회색 배경 위 스타일 텍스트 b), `t_b` (배경 이미지), `t_f` (배경 위 스타일 텍스트 b), `mask_t` (스타일 텍스트 b의 바이너리 마스크) 예시를 나타낸다.

2.6.2 Textctrl: Text Style Encoder

장면 텍스트 편집은, 텍스트 스타일(폰트, 색상, 공간 변환, 입체 효과 등)이 시각적으로 혼합되어 있어 스타일 특징 분리에 어려움이 있음. TextCtrl에서는 다중 작업 사전 학습(Text Style Disentanglement Pretraining)을 통해 세분화된 스타일 분리를 실현함.

앞서 생성한 Dataset을 바탕으로, 해당 인코더 학습을 실시함. 예시는 그림 2.4과 같다.

Figure 2.4: 왼쪽부터 순서대로 각각 `source_image` (입력 텍스트 이미지), `removal_image` (텍스트 제거 결과), `seg_image` (텍스트 분할 결과), `color_gt` (색상 전송 대상 텍스트), `color_image` (색상 전송 결과), `font_gt` (폰트 전송 대상 텍스트), `font_image` (폰트 전송 결과)를 나타낸다. 위의 그림은 첫 에폭시, 아래의 그림은 에폭 50이후의 이미지이다.

2.6.3 Textctrl: Glpyh Encoder

TextCtrl은 문자 수준 텍스트 인코더 T를 사용하여 대상 텍스트 임베딩을 시각적 글리프 구조와 정렬되도록 처리한다. 이 인코더는 입력 텍스트를 Transformer 기반 인코더에 통과시켜 글리프 구조 특징 C_{struct}를 생성한다. 이후 생성된 C_{struct}는 고정된(frozen) 상태의 사전 학습된 장면 텍스트 인식기(recognizer)로부터 추출된 시각적 특징과 의미적으로 정렬되며, 이 과정에서 **CLIP 손실** $\mathcal{L}_{\text{clip}}$이 사용된다.

본 모델에서 한국어 내용을 반영하기 위해, 기존의 영문 글리프를 인식하던 부분을, 유니코드를 사용하여 한글 '가'-'힣'까지 총 11,172자로 수정하여, 해당 인코더를 재학습했다.

2.6.4 Links

Github : https://github.com/Ea3124/SRNet-Datagen_kr

2.7 UI/UX 구체화

본 모델을 이용한 웹 어플리케이션 제작을 위해 UI/UX를 설계 및 구체화하였다.

Figure 2.5: 설계된 UI/UX 화면 중, 2가지 화면을 발췌한 이미지이다.

Figma(UI/UX): https://www.figma.com/design/JvbYnuSH3OT1lQWxcJpusq/TMOJI?node-id=67-2&t=Cm33fTZ9oVOoKKpk-1

References

Cui, C., Sun, T., Lin, M., Gao, T., Zhang, Y., Liu, J., Wang, X., Zhang, Z., Zhou, C., Liu, H. et al. (2025), 'Paddleocr 3.0 technical report', *arXiv preprint arXiv:2507.05595* .

Zeng, W., Shu, Y., Li, Z., Yang, D. and Zhou, Y. (2024), 'Textctrl: Diffusion-based scene text editing with prior guidance control', *Advances in Neural Information Processing Systems* **37**, 138569–138594.