TAS-LayerD: Text-Aware Layer Decomposition with Adaptive Instance Splitting

DLCV Final Project - Challenge 2: Layer Decomposition - Group 13
網媒所碩二 R13944023 鄭絜元, 生醫電資所碩一 R14945026 張詠哲,
資管四 B11303043 施漢樺, 工海所碩二 R13525070 尤梓薰

Abstract

Decomposing graphic design images is challenging due to complex occlusions and diverse text elements. We present a **LayerD-based** pipeline that integrates **EasyOCR** and **Hi-SAM** for precise text segmentation. By applying **adaptive dilation** and **instance-level layer splitting**, our method robustly separates text, objects, and backgrounds while preserving high visual coherence.

Method Overview

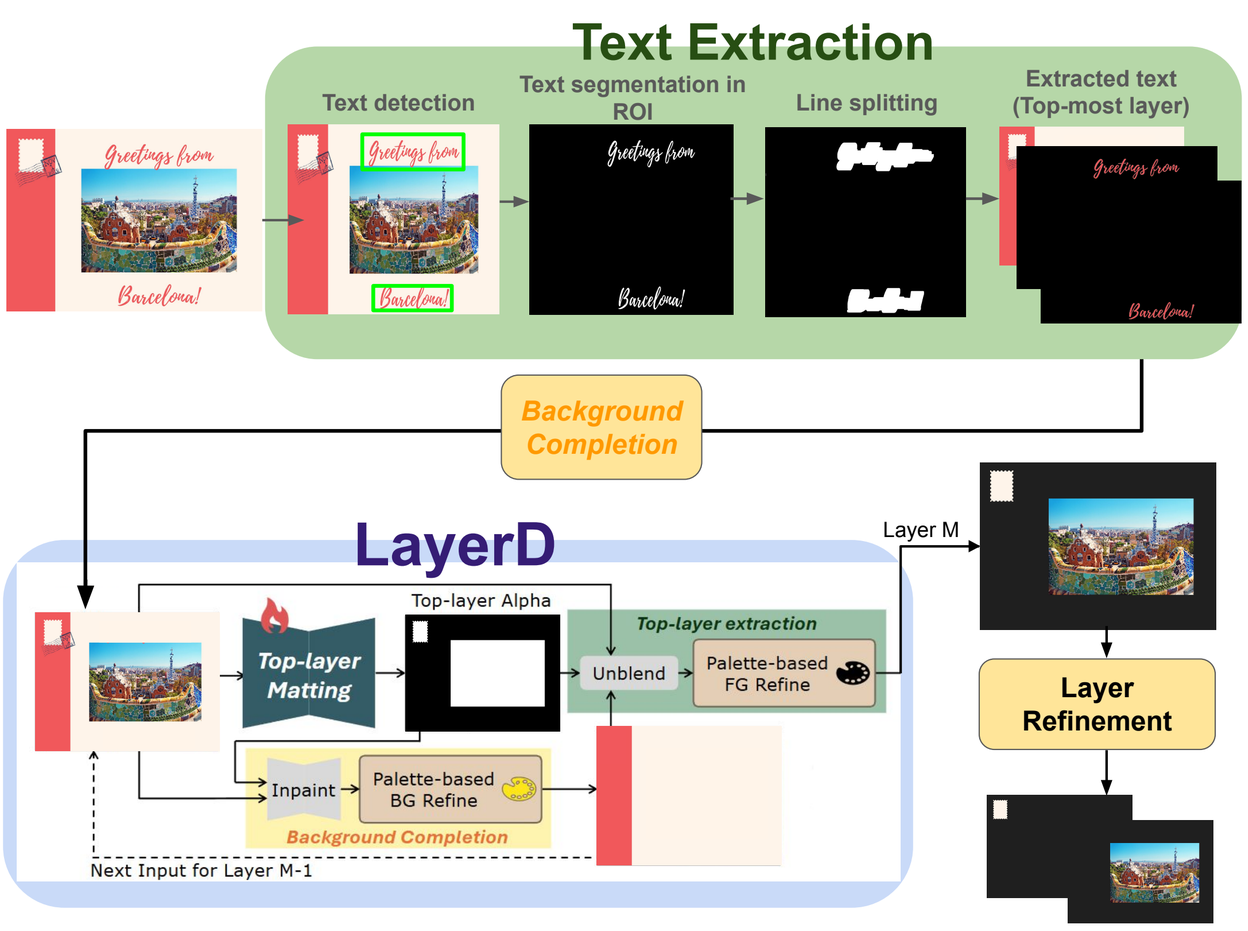

Figure 1: Model Architecture

Module 1: Text Extraction

We optimize the model for text extraction through the following three steps:

1. **Text Detection:** EasyOCR
2. **Text Segmentation:** Hi-SAM (original for large regions, patch-sliding for small regions)
3. **Line Splitting:** Adaptive dilation and contour analysis

Figure 2: Comparison of Text Regions with and without Hi-SAM.

Figure 3: Comparison of Text Segmentation

Module 2: Layer Decomposition

- **Matting:** We extract top-layer mask using the BiRefNet and train on our dataset
- **Inpainting:** We inpaint the background using LaMa and apply the Palette-based foreground and background refinement in LayerD

Module 3: Layer Refinement

We split over-merged layers predicted by LayerD into instance-level layers using **connected component analysis** and **feature-based KMeans clustering** while maintaining accurate image reconstruction.

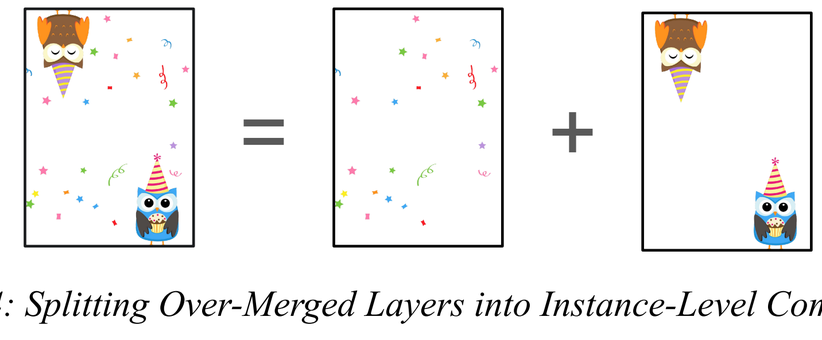

Figure 4: Splitting Over-Merged Layers into Instance-Level Components.

Experiments & Results & Ablation Study

Method	Evaluation Score
baseline (LayerD)	35.2
LayerD + layer splitting	39.68
Text extraction + LayerD (easyocr only)	39.96
Text extraction + LayerD (hi-sam only)	39.36
Text extraction + LayerD (easyocr + LayerD matting + line splitting)	50.35
Text extraction + LayerD (easyocr + Hi-SAM + line splitting)	64.70
Text extraction + LayerD + layer splitting **(ours)**	76.48

Table 1: All methods & evaluation score on Codabench.
Total score = 0.1 · global score + 0.9 · mean subset score

Figure 5: Comparison of LayerD and Our method

Conclusion

We successfully implemented a layer decomposition pipeline. By integrating **Text Extraction**, **Layer Decomposition** and **Layer Refinement**, we achieved a **117%** improvement in evaluation score for demonstrating superior performance on layer decompose tasks.

References

1. **[LaMa]** Suvorov et al., "Resolution-robust Large Mask Inpainting with Fourier Convolutions", *WACV*, 2022.
2. **[Hi-SAM]** Ye et al., "Hi-SAM: Marrying Segment Anything Model for Hierarchical Text Segmentation", IEEE TPAMI, 2025.
3. **[LayerD]** Suzuki et al., "LayerD: Decomposing Raster Graphic Designs into Layers", *ICCV*, 2025.
4. **[BiRefNet]** Peng Zheng et al. Bilateral reference for high-resolution dichotomous image segmentation. CAAI Artificial Intelligence Research, 3, 2024.
5. **[EasyOCR]** EasyOCR. JaidedAI. https://github.com/JaidedAI/EasyOCR