Synthetic Map Generation to Provide Unlimited Training Data for Historical Map Text Detection

Zekun Li\textsuperscript{1}, Runyu Guan\textsuperscript{2}, Qianmu Yu\textsuperscript{2}, Yao-Yi Chiang\textsuperscript{1}, Craig A. Knoblock\textsuperscript{2}

\textsuperscript{1}University of Minnesota
\textsuperscript{2}University of Southern California
Motivation

• **Automatically reading text** from map images could speed up map interpretation

• Many existing text detection models are trained with data from **different domains** (e.g. ICDAR scene images)

• Gathering **training data** for historical maps is important, while manual annotation takes a lot of time and effort
Related Work

• Text Detection Datasets
  • International Conference on Document Analysis and Recognition (ICDAR)
    • Has been releasing text detection datasets since 2011
    • Before 2015: focused scene images and born-digital images [Karatzas et al. (2013)]
    • After 2015: incidental scene images [Karatzas et al. (2015)][Nayef et al. (2019)]
  • COCO-Text [Veit et al. (2016)]
    • Machine-printed and hand-written text, multi-lingual
  • SCUT-CTW1500 [Liu et al. (2019)] & Total Text [Ch’ng et al. (2019)]:
    • Contain curved text instances
  • David Rumsey Map Collection [Weinman et al. (2019)]:
    • Historical map images
    • Dataset that we use for evaluation in this paper
Related Work

• Synthetic Data Generation
  • **SynthText** [Gupta et al. (2016)]
    • Large scale dataset with 800K images
    • Segmentation-based method for text label placement
  • **UnrealText** [Long et al. (2020)]
    • 600K images with 12 million word instances
    • Utilize Unreal 3D graphics engine for placing text on object surfaces
Approach

• We propose to generate **synthetic historical** maps images to aid the training of text detection models

• General Idea
  • Generate **synthetic map background** without any text labels
  • Automatically **place text labels** and generate ground-truth **annotation**
Synthetic Map Generation

- Public map sources such as OpenStreetMap (OSM) contain **separable** map layers: rasterized map **background** layer and **text** layer
  - Use clean (no text) OSM map background to produce style-transferred historical map
- Generative Adversarial Networks (GAN) can efficiently perform **style transfer**
  - CycleGAN model for the map style transfer – this can be replaced with other models as well
- Target map style
  - **Ordnance Survey** 6-inch map during year 1888-1913
Synthetic Map Generation - CycleGAN

Cycle Consistency Loss:
\[
\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1].
\]

Adversarial Loss:
\[
\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))].
\]
Sample Synthetic Map Images

input OSM map tiles

output synthetic historical map tiles
Text Layer Overlay

- Three types of geometries (QGIS PAL API for automatic label placement)
  - **Polygon**: text label on the polygon
  - **Line**: text label on the line
  - **Point**: text label near the point

- Font style and size
  - Roughly three levels of **font sizes**: large, medium and small
  - Collected 16 **historical** font styles
  - Same type of geo features have the **same** font size and style

<table>
<thead>
<tr>
<th>Groups</th>
<th>Size (pt)</th>
<th>Geo Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large</td>
<td>[60,80]</td>
<td>canal, city, county, town, village, waterfall, wetland, island</td>
</tr>
<tr>
<td>Med.</td>
<td>[35,45]</td>
<td>airfield, airport, allotment, archaeological battlefield, camp site, cliff, dock, farmland, farm, forest, fort, hamlet, nature reserve, reservoir, ruins, vineyard, rail river, stream, others (e.g. streets)</td>
</tr>
<tr>
<td>Small</td>
<td>[20,30]</td>
<td></td>
</tr>
</tbody>
</table>
Text Annotation Generation

- We provide following annotation information
  - Bounding polygon for each text region
  - Centerline across the text region
  - Local height (or diameter) of the text region

*Details for automatically computing the annotation info can be found in the paper*
Experiment Setting

• Text Detection Model: PSE-Net [Wang et al. (2019)]

• Training Strategies (notice that we did not use any real map data for training)
  • ICDAR: train from scratch on ICDAR 2015 dataset only
  • SynthMap: train from scratch on our SynthMap dataset only
  • ICDAR + SynthMap: pretrain on ICDAR 2015, fine tune on SynthMap dataset

• Evaluation Metrics [Wolf et al. (2006)]
  • Considers 1:1 matching, 1:N matching and N:1 matching of GT and prediction
## Experiment Results

<table>
<thead>
<tr>
<th></th>
<th>ICDAR2015</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>prec.</td>
<td>recall</td>
<td>F1</td>
<td>prec.</td>
<td>recall</td>
<td>F1</td>
</tr>
<tr>
<td>D0006</td>
<td>84.30%</td>
<td>44.80%</td>
<td>58.50%</td>
<td>68.90%</td>
<td>25.30%</td>
<td>37.00%</td>
</tr>
<tr>
<td>D0017</td>
<td>81.10%</td>
<td>49.30%</td>
<td>61.30%</td>
<td>85.30%</td>
<td>63.40%</td>
<td>72.70%</td>
</tr>
<tr>
<td>D0041</td>
<td>71.90%</td>
<td>48.90%</td>
<td>58.20%</td>
<td>71.70%</td>
<td>70.80%</td>
<td>71.20%</td>
</tr>
<tr>
<td>D0042</td>
<td>81.28%</td>
<td>34.18%</td>
<td>47.75%</td>
<td>75.88%</td>
<td>48.65%</td>
<td>58.83%</td>
</tr>
<tr>
<td>D0079</td>
<td>45.30%</td>
<td>4.20%</td>
<td>7.70%</td>
<td>40.50%</td>
<td>20.60%</td>
<td>27.30%</td>
</tr>
<tr>
<td>D0089</td>
<td>83.10%</td>
<td>49.90%</td>
<td>62.40%</td>
<td>75.90%</td>
<td>44.60%</td>
<td>56.20%</td>
</tr>
<tr>
<td>D0090</td>
<td>82.80%</td>
<td>55.80%</td>
<td>66.70%</td>
<td>90.60%</td>
<td>63.40%</td>
<td>74.60%</td>
</tr>
<tr>
<td>D0117</td>
<td>89.75%</td>
<td>56.13%</td>
<td>68.90%</td>
<td>72.72%</td>
<td>55.55%</td>
<td>62.95%</td>
</tr>
<tr>
<td>D5005</td>
<td>88.55%</td>
<td>57.68%</td>
<td>69.57%</td>
<td>78.38%</td>
<td>54.60%</td>
<td>64.23%</td>
</tr>
<tr>
<td><strong>All</strong></td>
<td>82.76%</td>
<td>45.00%</td>
<td>57.32%</td>
<td>74.90%</td>
<td>51.73%</td>
<td>60.62%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ICDAR + SynthMap</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>prec.</td>
<td>recall</td>
<td>F1</td>
<td>prec.</td>
<td>recall</td>
<td>F1</td>
</tr>
<tr>
<td>D0006</td>
<td>79.60%</td>
<td>27.70%</td>
<td>41.10%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0017</td>
<td>88.90%</td>
<td>60.80%</td>
<td>72.20%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0041</td>
<td>74.90%</td>
<td>72.35%</td>
<td>73.60%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0042</td>
<td>77.86%</td>
<td>55.75%</td>
<td>64.39%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0079</td>
<td>31.30%</td>
<td>13.60%</td>
<td>19.00%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0089</td>
<td>69.30%</td>
<td>48.10%</td>
<td>56.80%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0090</td>
<td>91.00%</td>
<td>70.30%</td>
<td>79.30%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0117</td>
<td>82.40%</td>
<td>55.12%</td>
<td>66.02%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D5005</td>
<td>82.55%</td>
<td>57.95%</td>
<td>67.87%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>All</strong></td>
<td>78.51%</td>
<td>55.25%</td>
<td>64.25%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ICDAR from scratch

SynthMap from scratch

ICDAR + SynthMap
Conclusion and Future Work

- We proposed a pipeline to generate unlimited amount of training data for map text detection algorithms
- We use style transfer model to convert OSM maps to historical style
- Text labels are placed according to the underlying geo feature locations
- This approach can be extended to generate synthetic data for word linking and road delineation
Demo and Code

• Live Demo


• Code released

https://github.com/zekun-li/generate_synthetic_historical_maps