An Accurate Threshold Insensitive Kernel Detector for Arbitrary Shaped Text

Speaker: Xijun Qian
We divide the detection into two parts, respectively detecting the text core and the text core area.
Label Generation

Decay Learning weight (DLW)

\[ ep = \frac{current\_epoch}{MAX\_epoch} \]

\[ TDLW = \lambda^{SM \times (1-ep)} \otimes \lambda^{ET \times ep} \otimes TM \]

\[ KDLW = \lambda^{SM \times (1-ep)} \otimes \lambda^{EK \times ep} \otimes KM \]

The white area is filled with 1, the gray area is filled with 0, the black area is filled with -1, and the shaded area is filled with the weight related to the text width.

Generate 7 label images for each training image
Loss Function

- **Loss function**
  - Total loss: $L = L_T + \lambda L_K$
    - **Loss of text area**: $L_T = BCE_T + Dice_T$
      - The cross entropy loss and Dice loss are:
        $$BCE_T = \frac{1}{|S|} \sum_{i \in S} [y_i \log x_i + (1 - y_i) \log(1 - x_i)] \otimes TDLW$$
        $$Dice_T = 1 - \frac{2|T_{\text{pred}} \cap T \cap TDLW|}{|T_{\text{pred}} \cap TDLW| + |T \cap TDLW|}$$
    - **Loss of text kernel**: $L_K = BCE_K + Dice_K$
      - The cross entropy loss and Dice loss are:
        $$BCE_K = \frac{1}{|km|} \sum_{i \in km} [y_i \log x_i + (1 - y_i) \log(1 - x_i)] \otimes KDLW$$
        $$Dice_K = 1 - \frac{2|K_{\text{pred}} \cap K \cap KDLW|}{|K_{\text{pred}} \cap KDLW| + |K \cap KDLW|}$$

- Use OHEM to deal with the imbalance of positive and negative samples
- Only count the pixels in the text area
Obtain the prediction of the text core area
Calculate the text core score and filter out areas with low scores
Expand the text core according to a fixed ratio to obtain the result

Key hyperparameters:
- Pixel threshold (PT)
- Box threshold (BT)
Some examples and results

![Image of some examples and results]

<table>
<thead>
<tr>
<th>Method</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextSnake [32]</td>
<td>82.7</td>
<td>74.5</td>
<td>78.4</td>
<td>1.1</td>
</tr>
<tr>
<td>TextField [2]</td>
<td>81.2</td>
<td>79.9</td>
<td>80.6</td>
<td>-</td>
</tr>
<tr>
<td>PSENet [1]</td>
<td>84.0</td>
<td>78.0</td>
<td>80.9</td>
<td>3.9</td>
</tr>
<tr>
<td>LOMO [34]</td>
<td>88.6</td>
<td>75.7</td>
<td>81.6</td>
<td>-</td>
</tr>
<tr>
<td>SPCNet [16]</td>
<td>83.0</td>
<td>82.8</td>
<td>82.9</td>
<td>-</td>
</tr>
<tr>
<td>CRAFT [14]</td>
<td>87.6</td>
<td>79.9</td>
<td>83.6</td>
<td>-</td>
</tr>
<tr>
<td>DB (800) [3]</td>
<td>87.1</td>
<td>82.5</td>
<td>84.7</td>
<td><strong>32</strong></td>
</tr>
<tr>
<td>TIKD (736)</td>
<td><strong>88.7</strong></td>
<td><strong>83.7</strong></td>
<td><strong>86.1</strong></td>
<td>16.3</td>
</tr>
</tbody>
</table>