1. WorkFlow

```
┌─────────────────────┐      ┌─────────────────────┐
│ text boxes generated│      │ text boxes generated│
│ by deep learning    │      │ by edge-based       │
│                     │      │ segmentation        │
└─────────────────────┘      └─────────────────────┘
```

combine

if number of   if number of
text boxes >1   text boxes <=1

```
┌─────────────────────┐      ┌─────────────────────┐
│     text boxes      │      │ use the MSER        │
│                     │      │ techniques          │
│                     │      │ to generate text    │
│                     │      │ boxes               │
└─────────────────────┘      └─────────────────────┘
                                        │
                                        ▼
                             ┌─────────────────────┐
                             │     text box        │
                             │                     │
                             └─────────────────────┘
```
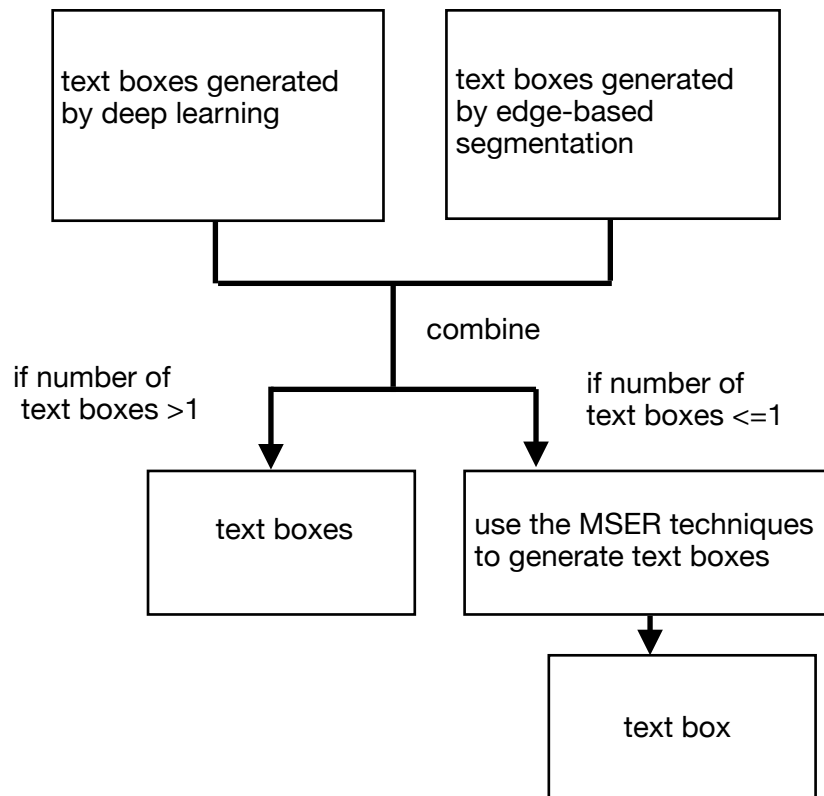
2. How dose edge-based segmentation identify text areas ?

Each character has strong edges. Therefore, we first detect all edges in an image, then filter them by sizes/ white pixel ratios. We then combine edge boxes in the same local neighborhood to form large text boxes.

Example

Step 1: Detect all edges (use canny edge detector) in an image. The detected edges areas are shown in the right panel.
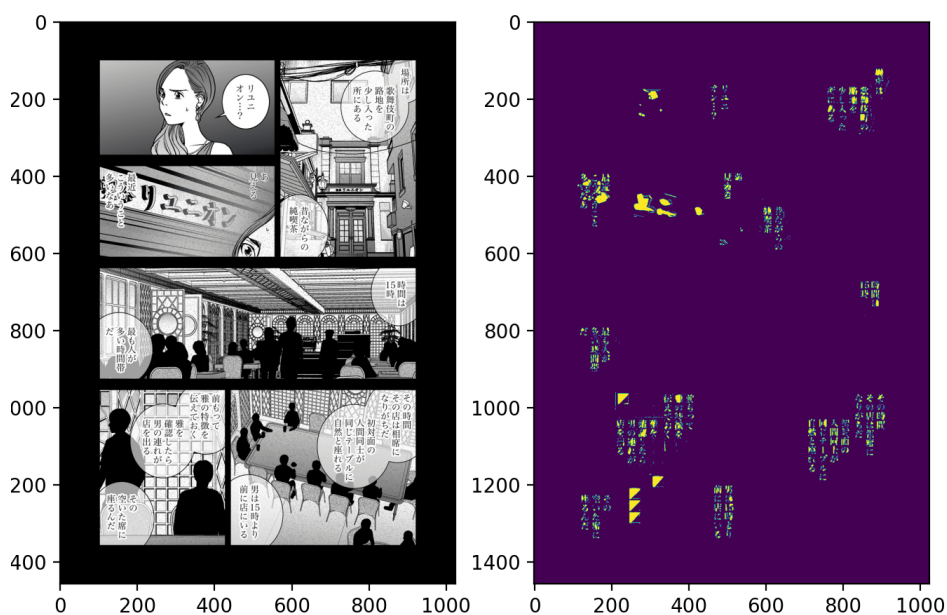
Step 2: group edges in the same local areas into larger text boxes
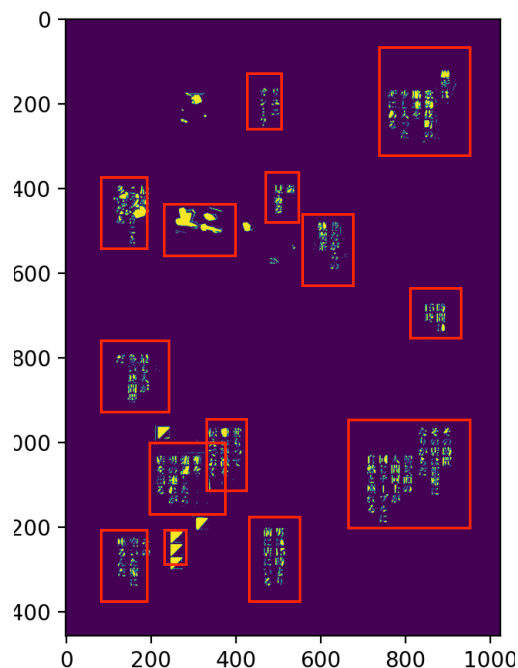
Step 3 : Since there are always spaces between neighboring characters or text lines, we check whether there exist spaces between neighboring edge areas. If no space, drop the text boxes. Also, if a box is a text box, it can be segmented by such space lines into smaller blocks whose widths or heights share some regularities. Therefore, we also segment each candidate box with vertical or horizontal lines into small blocks and check whether there is any regularity in their widths or heights. If no, then drop the box.

3. How to combine the results of the edge-based segmentation technique with the deep learning results ?

a. there are two types of boxes generated by the deep learning technique : bubble box and text box. For the bubble boxes having text boxes inside, drop the bubble boxes and keep only the text boxes. For the bubble boxes having no text boxes inside, drop them if their confidence scores are less than 0.6.
b. If the bubble boxes generated by the deep learning technique overlaps with the text boxes generated by the edge-based technique, drop the bubble boxes because the text boxes by the edge-based technique are usually tighter.
c. Concatenate the remaining boxes (including the bubble boxes and text boxes from the deep learning technique, and the text box from the edge-based technique) as the final result.

4. The original boxes from both the deep learning technique and edge-based technique usually suffer a risk of missing out some characters located in the same local areas. Therefore, it is necessary to expand the resulting boxes to reduce the risk. How to expand text boxes? We check whether there are any edge boxes near a text box, since edge boxes have a high probability to be text. If yes, expand the text box to include the edge boxes inside.

5. Why is MSER technique used ? There is a possibility that both edge-based technique and the deep learning technique returns no box at all. In this case, MSER is brought in to detect possible text boxes. How does MSER work? Text areas are usually 'stable' areas and MSER can

effectively detect these areas. We use MSER to detect all candidate areas in an image, then filter them by a number of metrics, such as extend, eccentricity, aspect ratio, etc.

6. Please note that the code has only been tested on images of which both sides are in the range of 1000-2000 pixels. The results may become worse on images of other sizes.