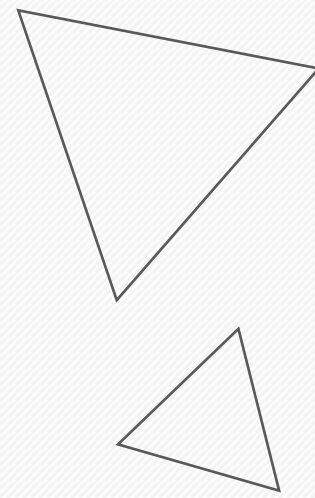


Detecting Oriented Text in Natural Images by Linking Segments

-----●

汇报人：盛驰云

●-----





文本检测： 即用单词或文本行的边界框定位文本，可以看作是应用于文本的目标检测。

文本特点： (1) 文本的高度宽度比值特别大或者小，
(2) 自然场景中的文本通常存在一定的旋转角度。



本文提出：引入旋转角度 θ 学习参数，回归参数 $(x,y,w,h) \rightarrow (x,y,w,h,\theta)$

Segment（段）：文本行的一部分（可以是字符或者文本行中任意某部分）

Linkng（连接）：用以连接每个Segment



Segments
(yellow boxes)



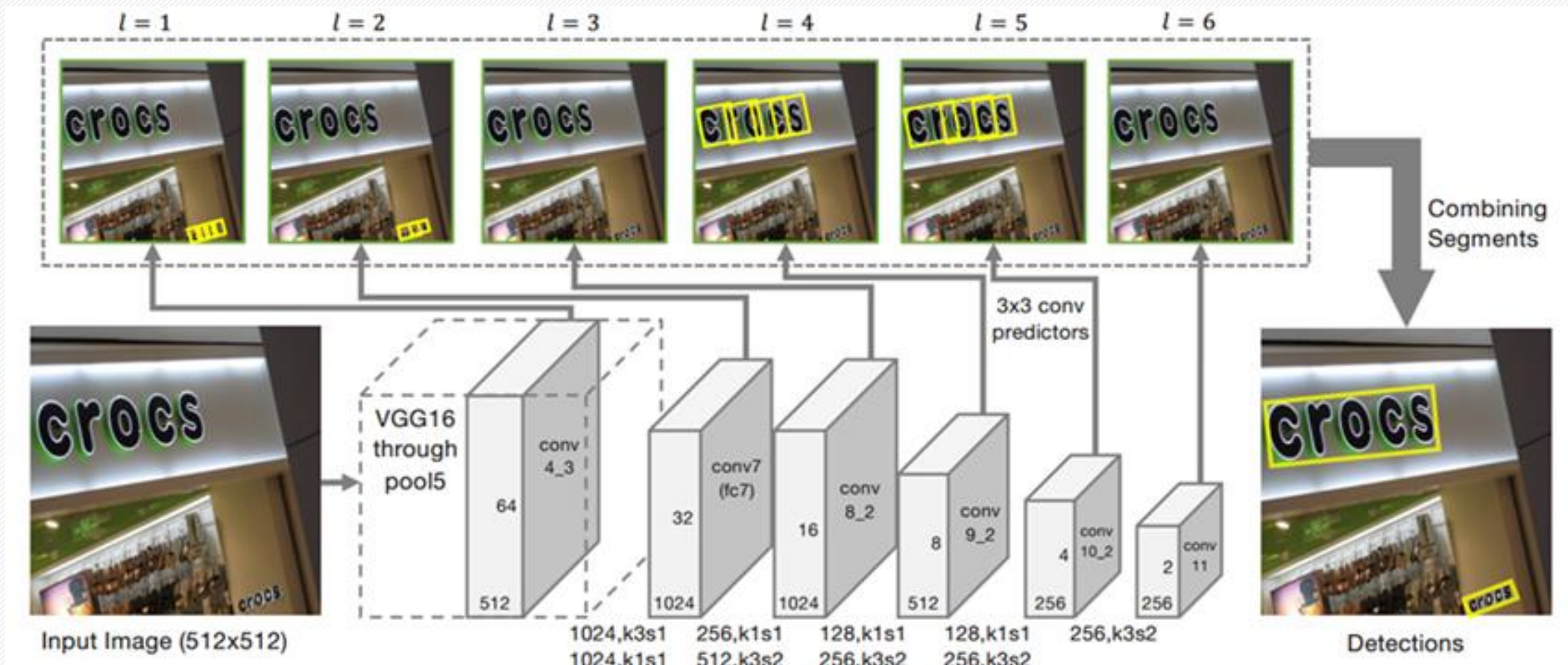
Links
(green edges)



Combined
detection boxes



conv4-conv11间的尺寸依次减少（每一层是前一层的1/2），从6个特征图上检测多尺度的Segment和Link



借鉴SSD思路，采用VGG16作为backbone进行特征提取



Segment类似于SSD中的回归box，表达形式如下：

$$s = (x_s, y_s, w_s, h_s, \theta_s)$$

default box个数，本文每个feature map的每个位置只采用了一个aspect ratio=1的default box，scale size设置结合当前层感受野：

$$a_l = \gamma \frac{w_I}{w_l}, \text{ where } \gamma = 1.5.$$

Segment计算公式：

$$x_s = a_l \Delta x_s + x_a$$

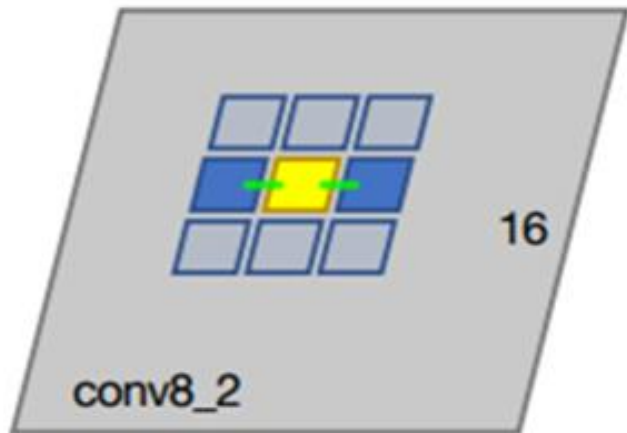
$$y_s = a_l \Delta y_s + y_a$$

$$w_s = a_l \exp(\Delta w_s)$$

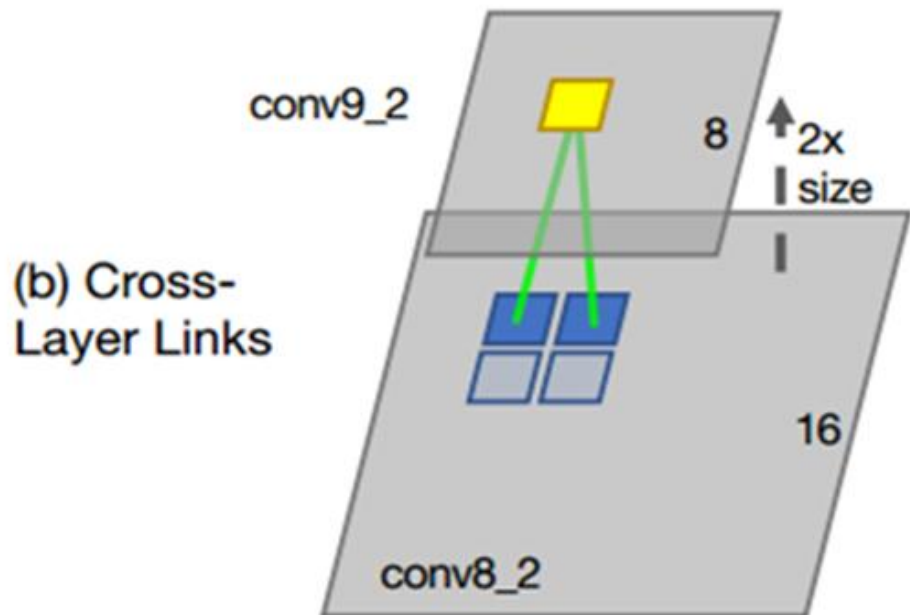
$$h_s = a_l \exp(\Delta h_s)$$

$$\theta_s = \Delta \theta_s$$

(a) Within-Layer Links



表示在同一层特征图里，每个Segment与8邻域内的Segment的连接状况，每个Link输出两通道，一通道是正分(两个Segment属于同一文本),另一通道是负分（两个Segment不属于同一文本）。每个predictor输出 $8 \times 2 = 16$ 维向量。

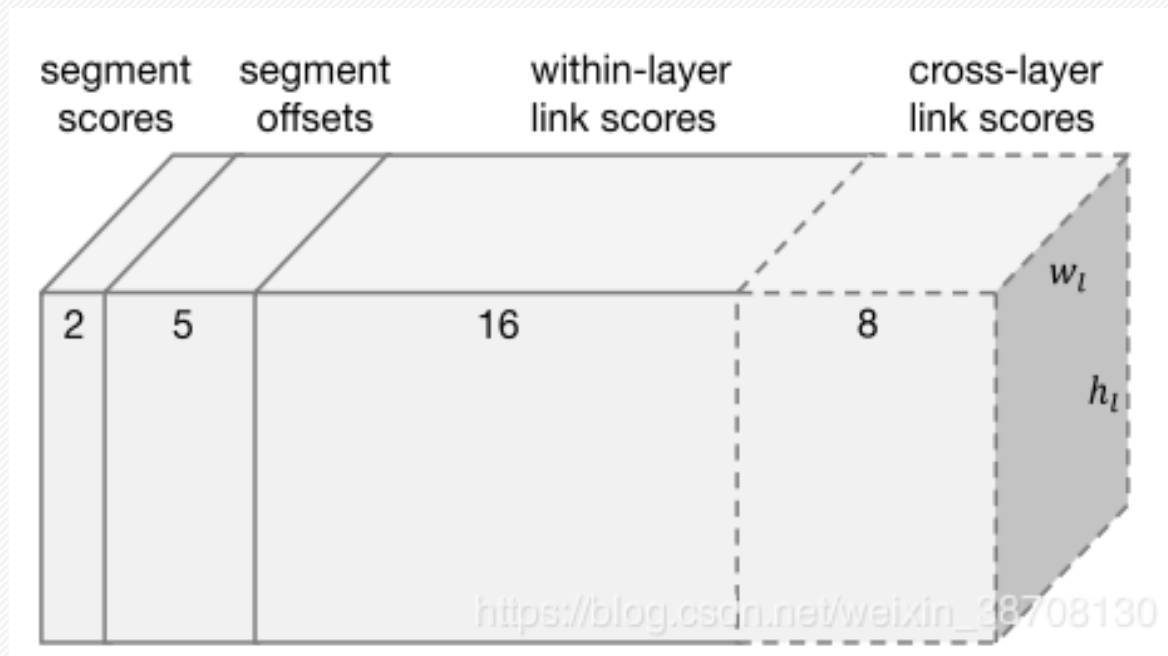


绿色的线代表cross-layer link 有连接（属于同一文本），后续combine算法中会将他们融合，即去除冗余。

cross-layer link连接了两个相邻特征图上的Segments。定义一个Segment的cross-layer邻居为前一层4邻域更小的segment，即前一层是后一层的邻居，但后一层不是前一层的邻居，故conv4_3的feature layer没有cross-layer邻居。



- (1) Segment的4个位置信息+旋转角度;
- (2) 每个Segment框内是否存在字符的分数;
- (3) 同层 (within-layer) 的每个Segment的Link的分数, 表示该方向是否有Link (共8个方向), 参数共 $2 \times 8 = 16$ 个;
- (4) 相邻层(cross-layer)之间也存在Link, 同样是该方向有Link还是没Link (共4个方向), 参数共 $2 \times 4 = 8$ 个。





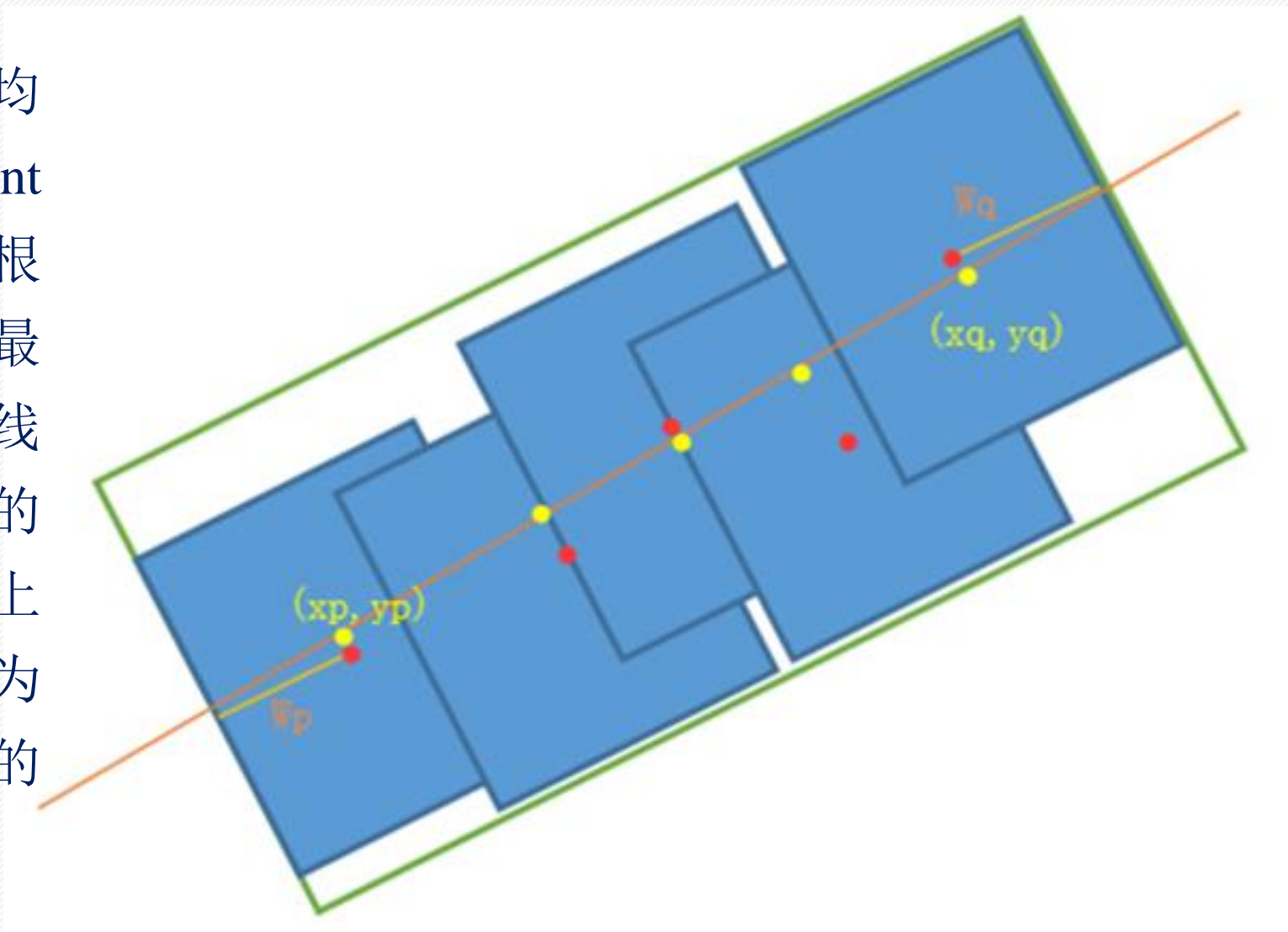
首先通过人工设定的 α 和 β (这两个值是采用网格搜索找到最优), 对网络预测的 Segments 和 Links 进行滤除, 将每个 Segment 看成 node, Link 看成 edge, 建立图模型, 再用 DFS(depth first search)找到连通分量, 每个连通分量包含一系列 Segments(用 B 表示), 用如下算法进行融合输出单词的 box。

Algorithm 1 Combining Segments

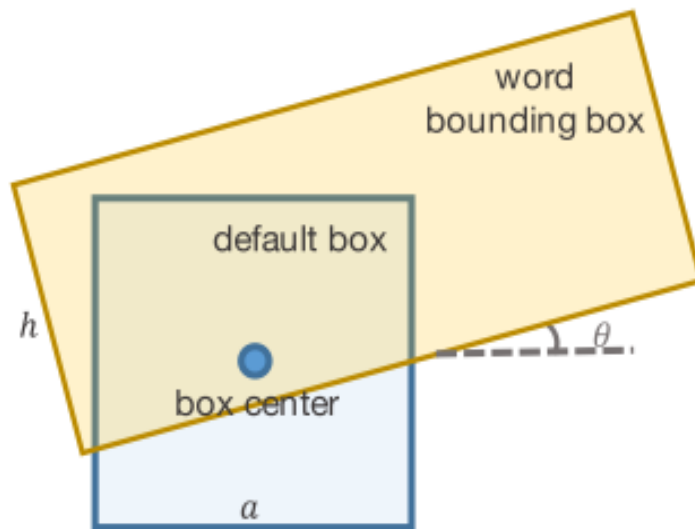
- 1: **Input:** $B = \{s^{(i)}\}_{i=1}^{|B|}$ is a set of segments connected by links, where $s^{(i)} = (x_s^{(i)}, y_s^{(i)}, w_s^{(i)}, h_s^{(i)}, \theta_s^{(i)})$.
 - 2: Find the average angle $\theta_b := \frac{1}{|B|} \sum_B \theta_s^{(i)}$.
 - 3: For a straight line $(\tan \theta_b)x + b$, find the b that minimizes the sum of distances to all segment centers $(x_s^{(i)}, y_s^{(i)})$.
 - 4: Find the perpendicular projections of all segment centers onto the straight line.
 - 5: From the projected points, find the two with the longest distance. Denote them by (x_p, y_p) and (x_q, y_q) .
 - 6: $x_b := \frac{1}{2}(x_p + x_q)$
 - 7: $y_b := \frac{1}{2}(y_p + y_q)$
 - 8: $w_b := \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} + \frac{1}{2}(w_p + w_q)$
 - 9: $h_b := \frac{1}{|B|} \sum_B h_s^{(i)}$
 - 10: $b := (x_b, y_b, w_b, h_b, \theta_b)$
 - 11: **Output:** b is the combined bounding box.
-



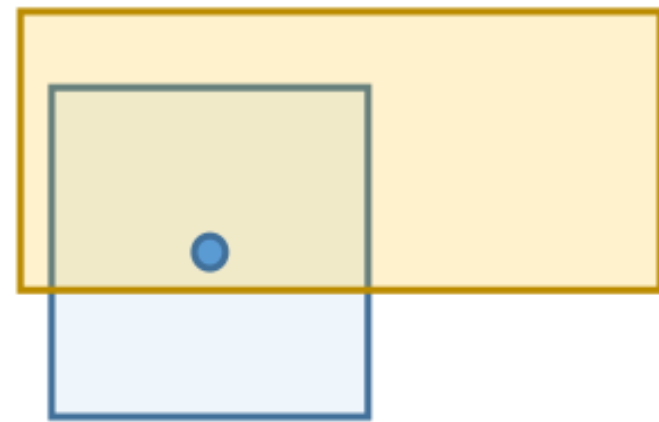
该算法其实就是一个平均的过程。先计算所有的Segment的平均 θ 作为文本行的 θ ，再根据已求的 θ 为已知条件，求出最可能过每个Segment的直线（线段，），以其中点作为word的中心点，最后用线段长度加上首尾Segment的平均宽度作为word的宽度，用所有Segment的高度的平均作为word的高度。



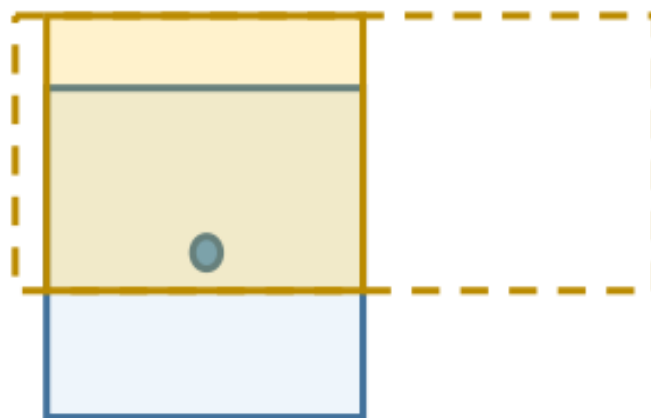
首先将文本行的 bbox 与 default box 进行水平对齐；然后对文本行 box 进行裁剪，保留 default box 与文本行相交的部分；最后再绕 default box 的中心点进行顺时针旋转，得到裁剪后的带角度的 bbox，即 ground truth Segment。网络要学习的偏移实际上就是 default box 相对于裁剪后的 bbox 的偏移。



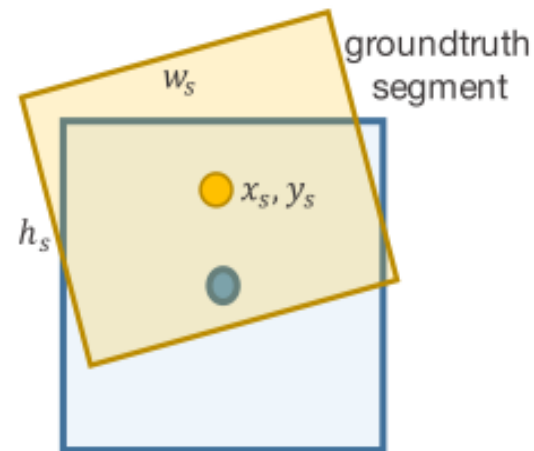
(1) Default box, word bounding box, and the center of the default box (blue dot)



(2) Rotate word clockwise by θ along the center of the default box



(3) Crop word bounding box to remove the parts to the left and right of the default box



(4) Rotate the cropped box anticlockwise by θ along the center of the default box

损失函数由三个部分构成，是否为文本的二分类Softmax损失，box的smooth L1 regression损失，是否为link的二分类Softmax损失。 λ_1 和 λ_2 控制权重，最后都设为1。

$$L(y_s, c_s, y_l, c_l, \hat{s}, s) = \frac{1}{N_s} L_{conf}(y_s, c_s) + \lambda_1 \frac{1}{N_s} L_{loc}(\hat{s}, s) + \lambda_2 \frac{1}{N_l} L_{conf}(y_l, c_l)$$

本文在标准ICDAR2015基准上，取得了75%的f-measure，相比同类算法大幅度提高了性能，在512x512图像上以超过20FPS的速度运行。此外，SegLink能够检测非拉丁文字的长行，例如中文等。





创新性： (1)Seglink可以检测多方向文本；
(2)Seglink将Link放入神经网络中学习，而不是像常规方法一样在后处理步骤中将多个bounding box通过合并算法合并；
(2)Seglink不受限于感受野，能够处理长文本甚至弯曲文本。

局限性： (1) α 和 β 阈值的设置需要人工设置。
(2)不能检测间隔很大、文字在图中比例很大的文本行，对不规则形状的文本检测能力有待提升。