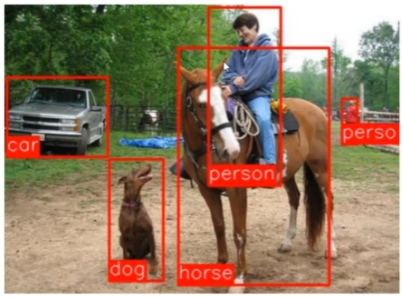
OCR识别原理：

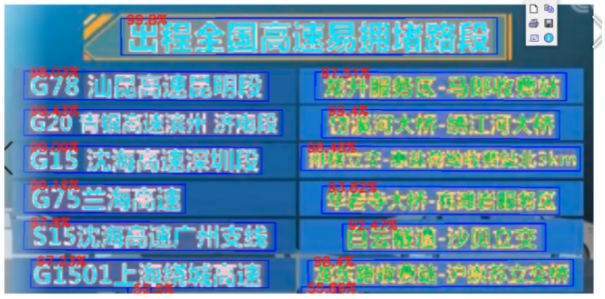
1. OCR（光学识别）
2. 如何从图像中提取文本：
    1. 检测到文字所在的位置（CTPN）

    2. 识别文本区域内容（CRNN）、

---

3. CTPN（Connectionist Text Proposal Network）
    1. 文本检测本质上属于物体检测，但是文本却跟常规的物体有较大的区别

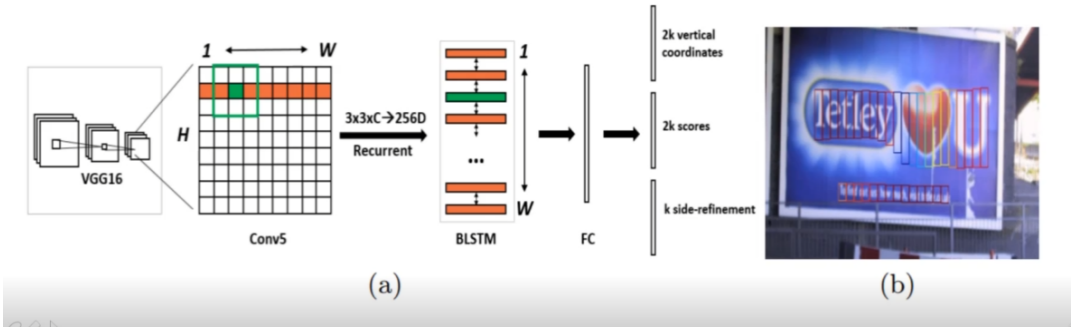




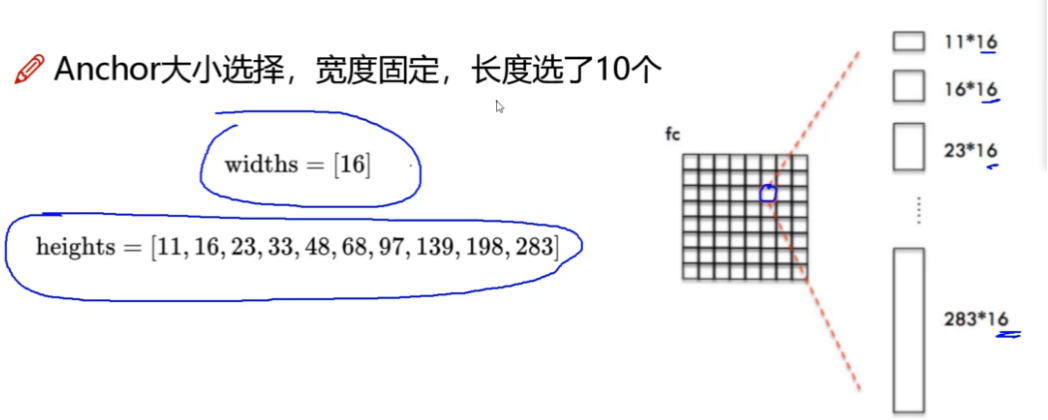
    2. 文字与传统的物体不太一样，基于实际的文字的组成来涉及检测框架

3. 文本通常从左往右写的(水平)，并且字之间的宽度都大致相同
4. 固定宽度，来检测文本高度即可，但是如何应对变长序列呢?
5. 本质上还是RPN方法，可将检测到的框拼接在一起

6. **VGG提取特征**，**BLSTM**融入**上下文信息**，基于RPN完成检测



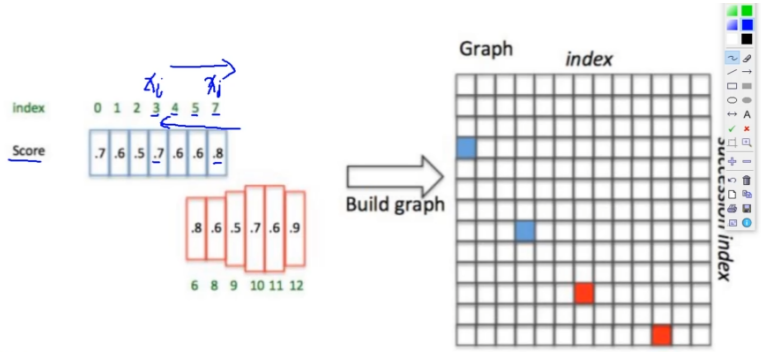
7. x不用再做微调，需要对框在边界上进行微调，只调位置，不调节宽度
8. Anchor大小选择，宽度固定，长度选择了10个



9. 输出结果包括三部分：
    1. 2k得分
    2. 2k回归
    3. 1k边界调整
10. 检测到每一个小块文本区域还需拼接成完整的文本区域
11. 规则，分前向和后向两部分：

12. CTPN网络：

举个栗子：



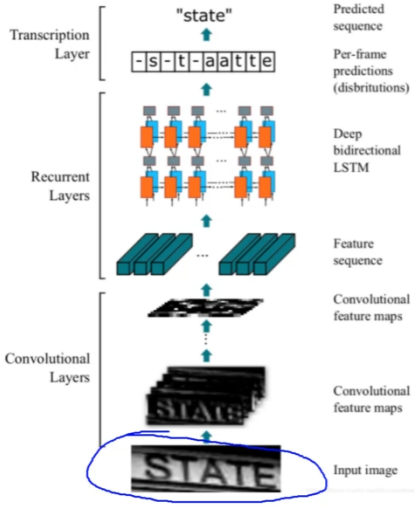
如果$score_i >= score_k$则这是一个长序列，反之则该序列被更长的序列所包含。

13. CRNN算法：

CRNN算法：

一张图解释了网络架构：

1. 首先CNN进行特征提取，接下来RNN进行序列特征提取，最后得出预测结果即可。



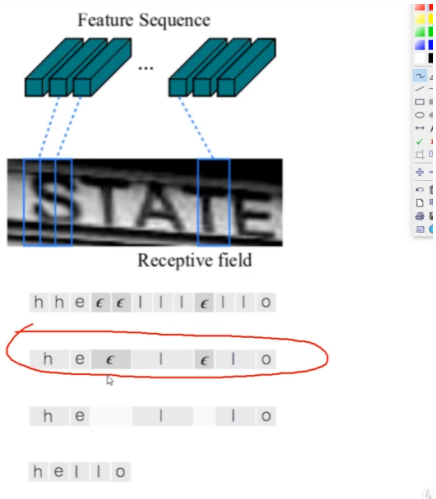
2. 首先CNN进行特征提取，接下来RNN进行序列特征提取，最后得出预测结果即可
3. 输入输出结果是变长的
4. 构建RNN的输入特征序列
5. 其中还涉及CTC模块，目的是对齐输入和输出结果

CRNN算法：

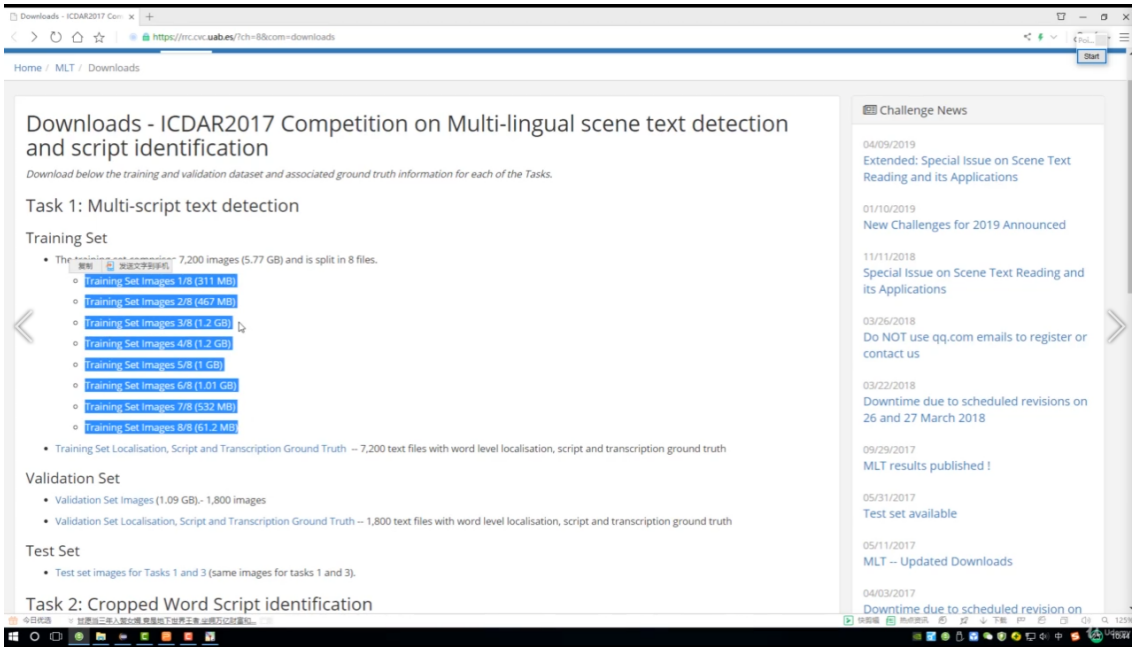
构建RNN的输入特征序列：

6.

其中还涉及了CTC模块，目的是对齐输入和输出结果：



4. OCR项目实战



https://vdn3.vzuu.com/HD/994512da-2a1d-11eb-afd1-8627592acda4.mp4?disable_local_cache=1&auth_key=1629107005-0-0-1649ac96894dadff9cd66262e2f846d1&f=mp4&bu=http-com&expiration=1629107005&v=tx