

Density Map Guided Object Detection in Aerial Images

1.论文解析

1.1 研究动机

1.1.1 目标检测面临的挑战

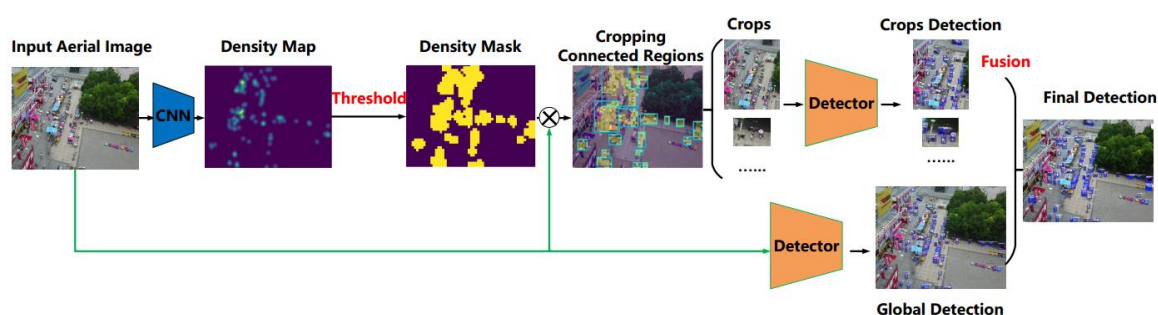
1. 1)目标大小变化很大，2)目标分布不均匀
2. 基于深度学习的架构在特定图像(如航空和医学图像)上不能产生令人满意的结果。
3. 早期的航拍图像目标检测工作[9,2]只是简单地利用了一般的目标检测架构，重点是提高对小目标的检测，但在实际应用中远不能令人满意。
4. 应用目标检测之前将原始图像裁剪成小块这个方式不能利用语义信息进行裁剪，因此导致大多数裁剪只有背景

1.1.2 本文的方法解决思路

1. 探索物体密度图在生成航空图像目标检测的裁剪图像方面的强大功能
2. 提出了一种基于密度图的航空图像检测框架——DMNet

1.2 方法解析

先训练密度图生成网络来预测每个航拍图像的密度图，然后，我们在生成的密度图上应用滑动窗口来收集像素强度之和，并将其值与密度阈值进行比较，形成密度蒙版。我们将像素强度大于密度阈值的窗口连接起来生成图像裁剪。最终的检测结果将图像裁剪后的检测结果与原始图像进行融合



1.2.1 Density map generation network（真实目标密度图）

1. 采用了 MCMM 《Singleimage crowd counting via multi-column convolutional neural network.》
2. 损失函数: 基于逐像素的平均绝对误差

$$L(\Theta) = \frac{1}{2N} * \sum_{i=1}^N \|D(X_i; \Theta) - D_i\|^2.$$

3. 参数含义

Θ 为密度图生成模块的参数

X_i 是输入的图像

$D(X_i; \Theta)$ 网络生成密度图

D_i 真实密度图

4. 不会引入采样层，因为在训练过程中，特征映射的大小大大增加，这可能会导致大分辨率图像的内存问题

1.2.2 Ground truth object density map

1. 采用了高斯核，固定核中是固定的，但在自适应核中采用 k -最近邻(KNN)方法计算。倾向于使用固定核

1.2.3 Improving ground truth with class-wise kernel

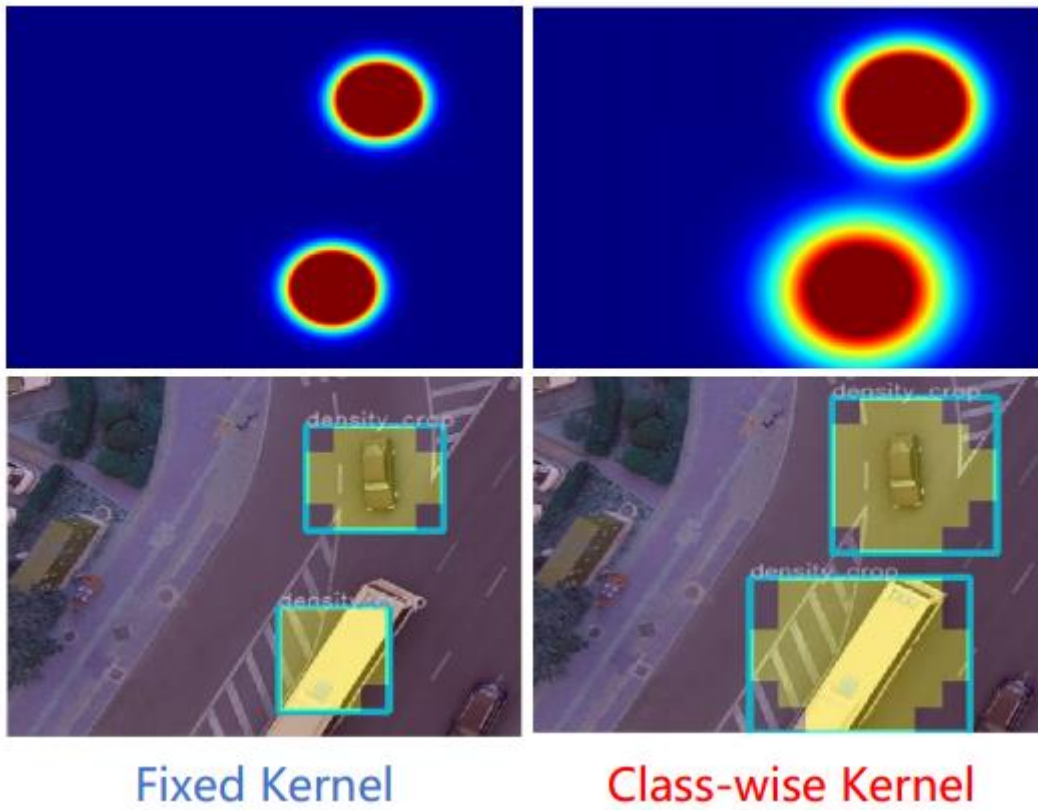
1. 为了解决可能出现的截断问题，我们提出了基于类的密度图地面真值生成方法。
2. 首先，对训练集进行探索性数据分析，分析每个目标类别的平均尺度，
3. 然后通过估计每个对象类别的平均尺度来计算 σ
4. 用该方程计算平均尺度，记录下每个类别的 σ 值，导入方程 2 以生成密度图

$$\sigma_i = \frac{1}{2} \sqrt{H_i^2 + W_i^2}.$$

eq2:

$$F(x) = \sum_{i=1}^N \delta(x - x_i) \times G_{\sigma_i}(x), \text{ with } \sigma_i = \beta \bar{d}_i, \quad (2)$$

其中， \bar{d}_i 是最邻近目标距离平均值， $G_{\sigma_i}(x)$ 是高斯核



类内核效果好于固定内核，因为解决了截断问题

1.2.3 Image cropping based on density map

1.2.3.1 Density mask generation

Algorithm 1 Density mask generation

Input: Aerial image Img . Density map Den . Sliding window size W_h, W_w . Density threshold TH .

Output: Density mask M .

▷ Initialization.

$I_h, I_w = Img.height, Img.width$.

$M = \text{zeros}(I_h, I_w)$

▷ Generate density mask

for h in $range(0, I_h, W_h)$ **do**

for w in $range(0, I_w, W_w)$ **do**

$S = \text{sum}(Den[h : h + W_h, w : w + W_w])$

if $S > TH$ **then**

$M[h : h + W_h, w : w + W_w] = 1$

end if

end for

end for

return M

算法如图所示

1. 初始化 航空图像 Img 作为 input , 密度图为 Den , 二维数组, 窗口的宽和高, W_h 和 W_w , 还有阈值 TH
2. 初始化 $Mask$, 高宽为 I_h, I_w
3. 初始化 $Mask$
4. 两个 for 循环, 让窗口在密度图里面扫描, 当窗口中的密度值之和大于阈值, 就给 $Mask$ 中对应的区域赋值为 1,
5. 最后输出 $Mask$

1.2.3.2 Generating density crops from density mask

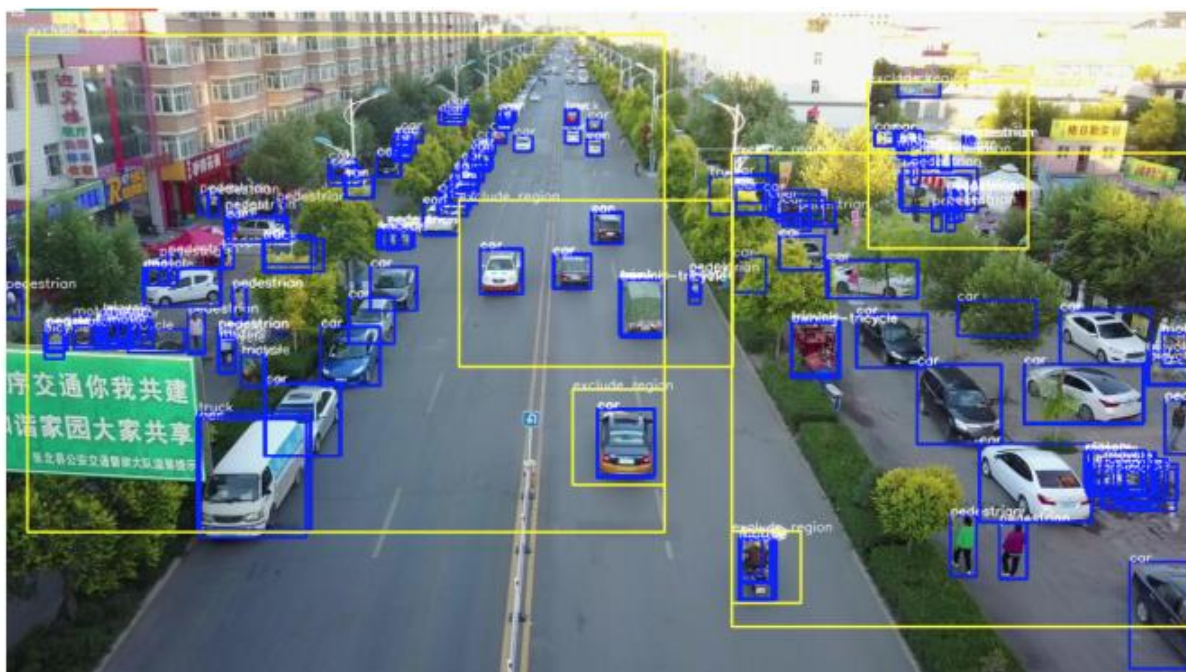
1. 首先, 选择密度掩码值为 “1” 的所有像素。
2. 接着, 将八个相邻的连通像素合并成一个大的候选区域。

- 最后，利用候选区域的限定矩形对原始图像进行裁剪，过滤掉分辨率低于密度阈值的图像裁剪 (crop)

1.2.4 Object detection on density crops

检测目标，并融合密度图像切片和整个图像的结果。

- 首先对原始验证集和密度图像切片分别进行检测
- 收集密度图像切片检测得到的预测边界框
- 将其与原始图像的检测结果进行融合
- 对所有边界框进行非最大抑制(NMS)，并计算最终结果
- 求值期间保留检测到的边界框。



问题 1：为什么需要对原始验证集进行检测，是不是为了生成图中黄框，将其与密度图切片蓝框进行对比

2. 实验设计

2.1 实验模型

1. 使用 MCNN[29]作为密度生成主干
2. Faster RCNN[17]作为检测器

2.2 消融实验

1. 验证密度阈值的影响：
2. DMNet vs uniform crops
3. density crop detection vs original image