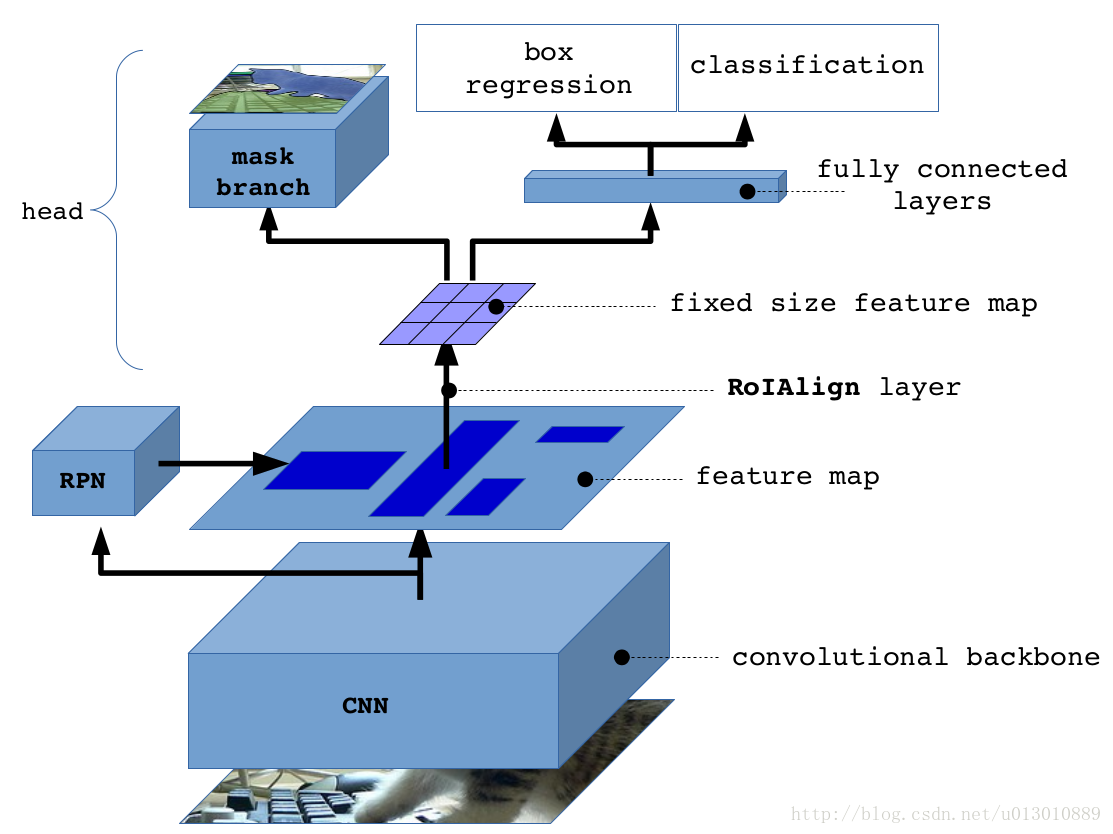


# 实例嵌入：没有建议的分段

在本文中，我将回顾实例分割领域的3篇论文。它们不同于主流的基于提议的快速RCNN方法，比如or和最新的，在多个数据集上获得最新的结果（，）。请参见有关蒙版RCNN的教程。

基于提议的实例分割体系结构有三个基本缺陷。首先，两个对象可以共享同一个边界框，或非常相似的框。在这种情况下，遮罩头无法判断要在框中拾取哪个对象。对于边界框中填充率低的线状对象（例如自行车和椅子），这是一个严重的问题。其次，架构中没有任何东西阻止像素在两个实例之间共享。第三，实例的数量受网络处理的建议数量（通常是数百个）的限制。

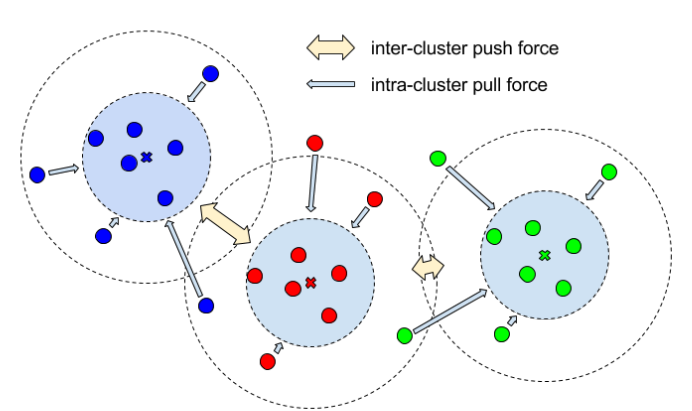


更重要的是，架构是复杂的，很难调整和“调试”。在目标检测这一问题的前兆中，已经有人成功地使用了更简单的单阶段体系结构，例如。

通过实例嵌入，在n维空间中为每个对象指定一个“颜色”。网络处理图像并产生与输入图像相同大小的密集输出。网络输出中的每个像素都是嵌入空间中的一个点。属于同一对象的像素在嵌入空间中比较接近，而属于不同对象的像素在嵌入空间中比较远。分析图像嵌入空间涉及到某种聚类算法。

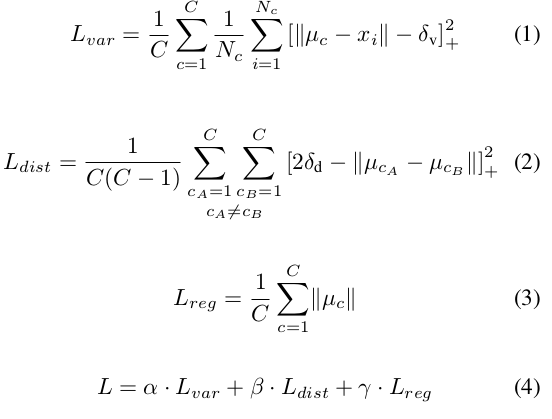
### 论文1：基于判别损失函数的语义实例分割

,



损失。本文使用了由三部分组成的对比损失函数：

1. 拉力。惩罚同一实例中所有元素与其平均值之间的距离。也就是说，获取一个实例的所有像素并计算它们的平均值。拉力将把同一实例的所有像素嵌入绘制到同一点。简而言之，减少每个实例嵌入的方差。
2. 一种推力。把所有的中心点（在嵌入空间，而不是空间中心）推得更远。
3. 正规化。中心不应该离原点太远。



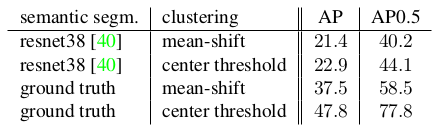
Alpha和beta与值1一起使用，gamma设置为0.001。两个三角形都是拉力和推力的阈值。

正在分析。在得到语义分割图（car，dog，computer，…）后，我们将每个类掩码细分为实例。这是通过在语义掩码中选择一个随机的未分配点并迭代应用mean-shift算法来找到实例的平均点来完成的。

平均值的第一个假设是初始选取的随机像素的嵌入。然后在该点周围（嵌入空间中）展开一组点，然后再次计算它们的平均值，并重复该过程，直到平均值的变化不显著为止。根据我的经验，该算法收敛的迭代次数不超过10次。大多数时候3-4次迭代就足够了。

嵌入空间中用于扩展实例掩码的半径与拉阈值相同。理论上，如果测试误差为0，并且中心之间的最小距离至少是方差分量拉阈值的两倍，我们可以使用这些阈值来解析图像。距离不大于拉阈值的所有点应属于同一实例。由于测试误差几乎从不为0，所以使用均值漂移算法来寻找嵌入的高密度部分的中心。

误差源

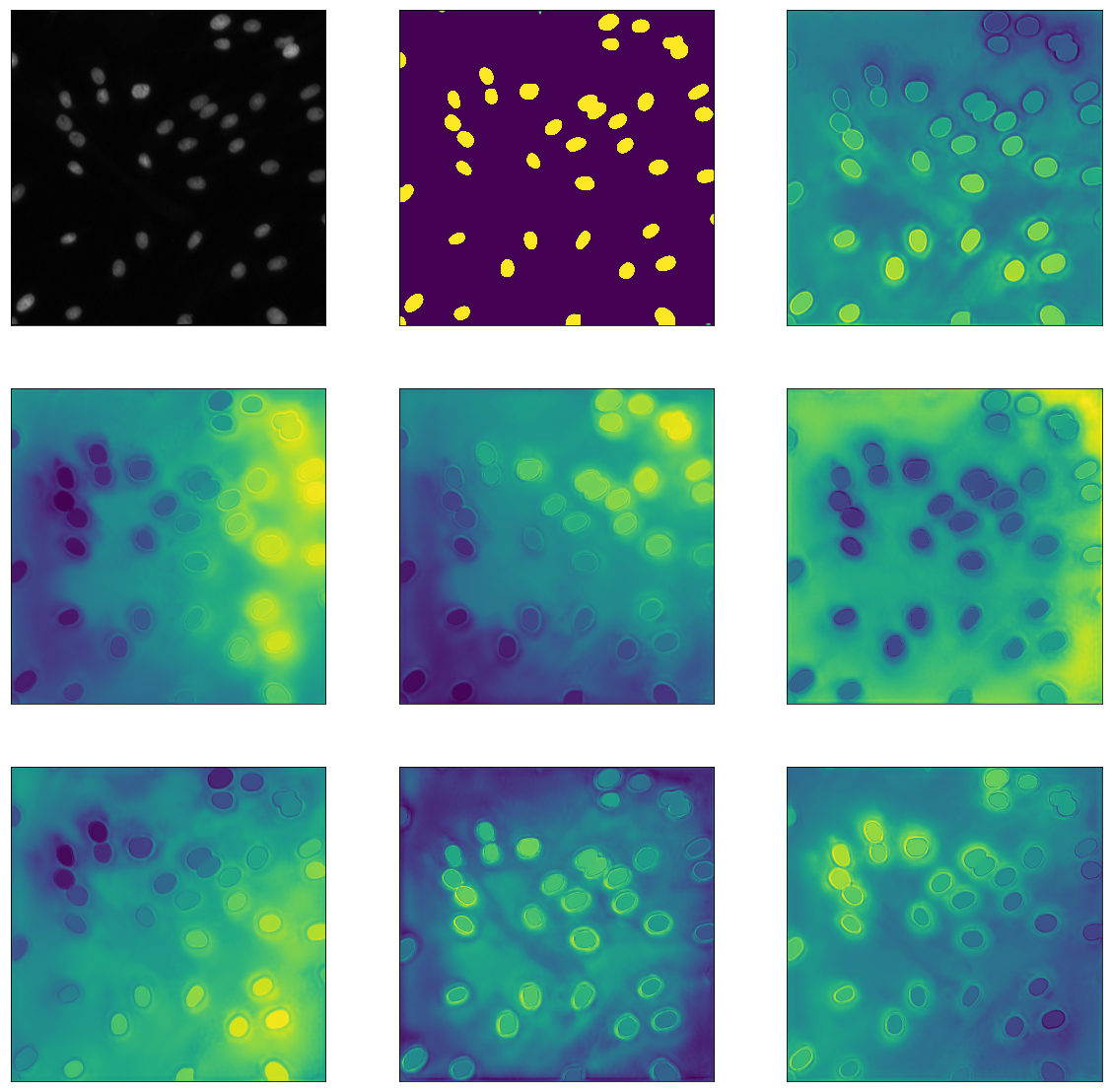


这些结果显示了大部分误差来自城市景观数据集。如果不进行语义分割，而是使用基本真理，则AP50结果从40.2跳到58.5。如果同时使用实际中心，而不是使用均值漂移进行估计，则得分几乎再增加20分，达到77.8分。未经COCO预训练的最新研究结果为57.1（见）。与使用语义切分基本真理相同。我们了解到嵌入本身可能非常好。

### 示例嵌入

下面是一个实例嵌入由一个网络训练产生，您的真实。它用于解决当前运行在Kaggle上的由提出的问题。目的是在医学图像中寻找细胞核。

左上角的图像是原始图像。中顶图像是语义分割（这里只有两类，背景和前景）。其余的图像是64个嵌入空间的前7个通道。从嵌入中可以明显看出，网络学习通道在空间上区分了细胞核。例如对角线或水平编码。一些编码到图像中心的距离。但是，在实例中，颜色是均匀的。这让我们对网络如何学习分段实例有了一些了解。



### 论文2：基于深度度量学习的语义实例分割

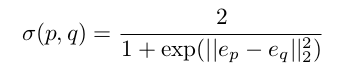
, , ,

本文的主要贡献是为每个像素学习一个种子度分数。分数告诉我们像素是否是一个很好的候选来扩展一个遮罩。在前面的论文中，种子是随机选择的，然后使用均值漂移算法对中心进行细化。在这里，只进行了一次扩展。



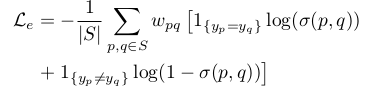
本文提出为每个像素学习几个可能的种子。我们为每个半径（嵌入空间）和类学习一个种子。因此，如果我们有C类，我们学习T带宽（半径），我们有每个像素的CxT种子“建议”。对于每个像素，只考虑得分最高的建议。

嵌入损失。在本文中，嵌入是对像素的惩罚。我们考虑相同实例的对和来自不同实例的对。

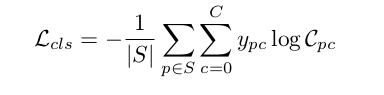


本文利用改进的logistic函数将嵌入空间中的欧几里德距离变换为[0，1]域。嵌入空间中较近的对将由函数分配一个接近1的值，较远的对将接近0。

自然地，logloss被用作损失函数。实例大小可能会有所不同，因此，为了缓解这种不平衡问题，会根据它们所属实例的大小对它们进行加权。



种子丢失。对于每个像素，模型学习几个种子分数。带宽（嵌入空间中的半径）和类的每个组合得一分。由于种子度得分很接近但与语义分割不一样，因此每次评估嵌入时都会确定每个种子的基本真实度。遮罩是围绕像素的嵌入展开的，如果具有基本真实实例的IoU超过某个阈值，则该像素被视为实例类的种子。这场失利将惩罚这个级别的低种子分数。



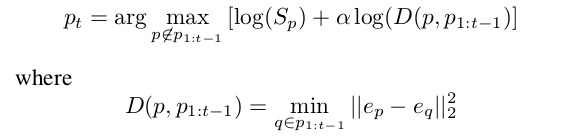
每批图像中只有10个左右的种子被评估，随机选取。学习了几个这样的模型，每个带宽一个。带宽越宽，物体就越大。在某种程度上，获得最高分数的带宽是模型将其估计值传递到实例大小（相对于嵌入空间中的距离）的方式。

训练过程。本文使用了在COCO数据集上预训练的ResNet-101骨干网。训练从没有分类/种子预测开始，即λ=0，随着嵌入更加稳定，训练进展到0.2。



在不同尺度（0.25、0.5、1、2）下对骨干进行评价，并将连锁结果反馈给种子和嵌入头。

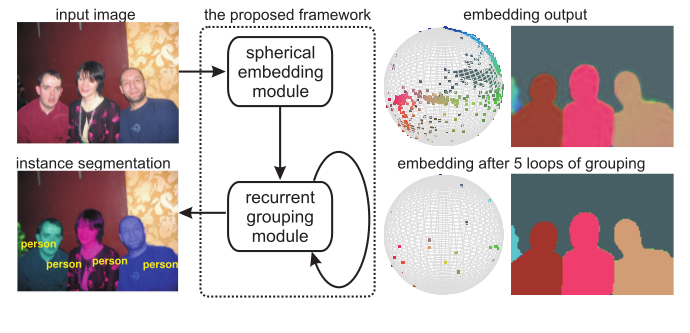
正在分析。自从种子学会了这个过程就很直接了提出了一种图像最佳种子集的选取方法它一方面优化了种子度得分，另一方面优化了嵌入空间的多样性。



迭代地选择种子，每个新种子被选择为与先前选择的种子在嵌入空间中的距离。选择的第一个种子是图像中种子得分最高的像素。第二种是种子，一方面种子得分高，另一方面种子在嵌入空间不紧密。使用参数alpha控制两个需求之间的平衡。Alpha是一个需要调整的参数，这个参数的测试范围在0.1到0.6之间。与NMS不同，它鼓励嵌入空间的多样性，而不是空间多样性。

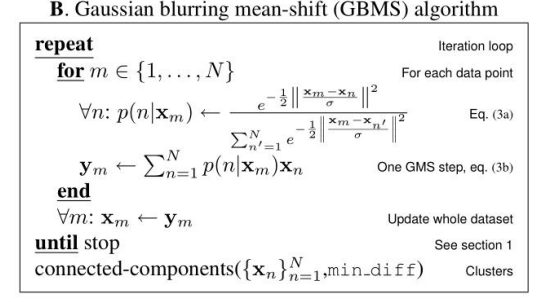
### 论文3：以分组为例的递归像素嵌入

,



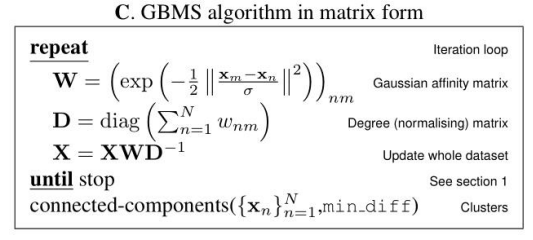
本文提出了在N球上嵌入，并利用余弦距离来测量像素的接近度。然而，本文的主要贡献是基于改进的高斯模糊均值漂移（GBMS）算法的递归分组模型。

GBMS是一种迭代算法，类似于第一篇论文中用于寻找实例中心的简单mean shift算法。在这个版本中，所有像素都被认为是潜在的种子。所有像素在每次迭代时都会根据其周围的密度进行更新。向“重心”移动，就好像图像的嵌入空间是一个产生行星的星云。点之间的距离越远，它们对彼此的影响就越小。距离是由高斯函数的带宽控制的，这是标准差，下面的算法就清楚了。



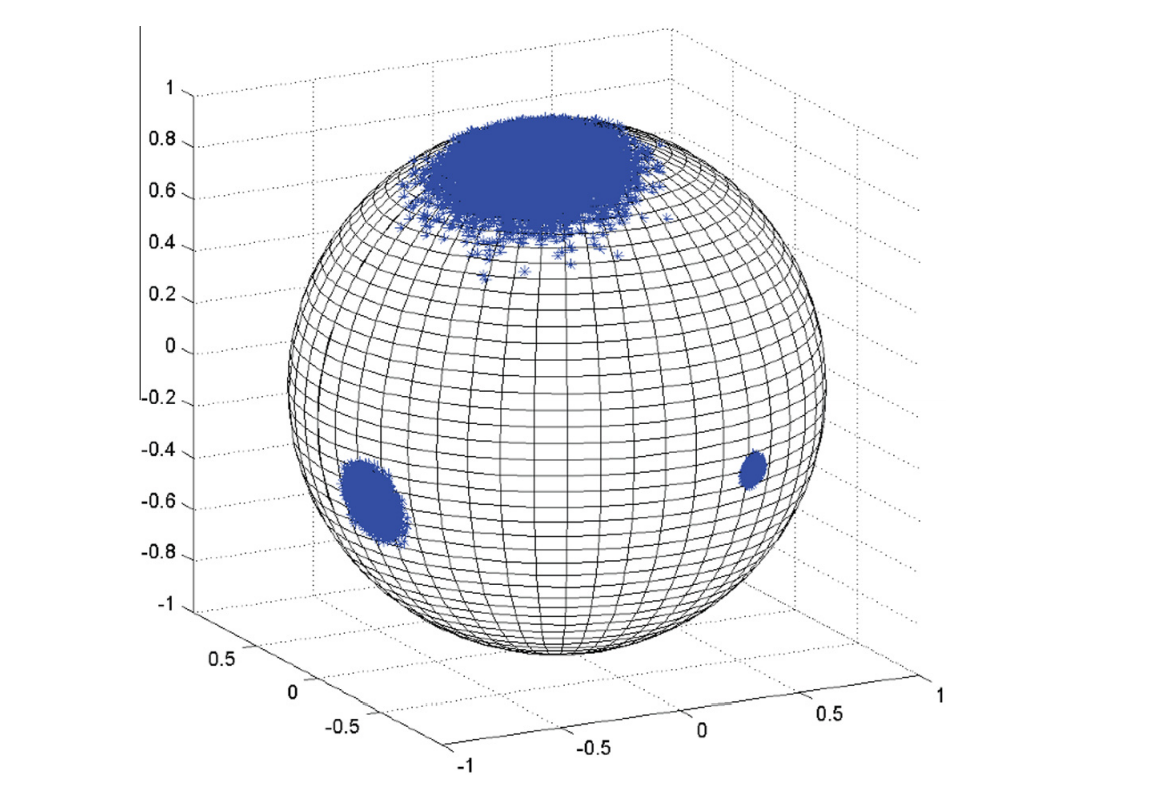
对于GBMS，有三次收敛的保证，因此在多次应用变换之后，我们最终应该得到非常密集的、几乎是点状的簇。有关GBMS的更多信息，请参阅。

为了将该算法融入到网络中，该算法使用矩阵运算来表示。



简单地应用上述算法是没有意义的，因为嵌入是在球体上，并且它们的邻近性是用余弦变换来测量的。描述所有点之间距离的关联矩阵使用以下转换计算：

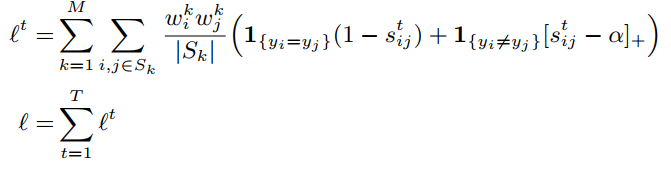
在球体上测量距离，而不是使用L2范数。此外，在应用GBMS步骤之后，需要规范化生成的嵌入，以便它们位于单位球体上。



训练。使用成对像素损失，类似于前面的文章，其中对不同对之间所需的距离设置了阈值（alpha）。使用范围为[0,1]而不是[-1，-1]的校准余弦距离对每对进行评估。



损失通过递归分组模型的每个应用程序进行反向传播。后期的应用只会出现非常困难的情况。例如，作者将此特性与用于快速RCNN训练的硬负挖掘进行了比较。



作者在论文中使用0.5作为alpha的值。请注意，实例的大小用于重新平衡大小实例之间的损失。

正在分析。经过分组模块的多次应用，集群应该非常密集，随机选取值应该产生足够好的种子。

出于实际目的，只使用GBMS步骤中的一些像素是有意义的，因为计算相似性矩阵可能会被证明是非常昂贵的。像素的数量是速度/精度权衡的考虑因素。

### 其他方法

实例嵌入并不是基于方案的网络的唯一选择。下面是一些使用其他方法解决实例分割问题的论文

* 重复注意的端到端实例分割
* 深分水岭变换实例分割
* 关联嵌入：用于联合检测和分组的端到端学习
* SGN：用于实例分割的顺序分组网络

### 摘要

与基于建议的解决方案相比，这些论文的结果没有竞争力。我们回顾了3篇关于损失和解析的不同解决方法的论文。

1. 语义实例分割采用了非成对损失函数，具有判别损失函数。使用图像中的所有像素生成更丰富的渐变。
2. 通过深度度量学习进行语义实例分割，引入种子模型，帮助我们在分类的同时选择最佳的种子，优化速度。
3. **Recurrent Pixel Embedding for Instance Grouping** GBMS, a variant of mean-shift, was used inside the network in both training and parsing. Creates very dense clusters.

These approaches could probably be combined and refined to produce far better results. They are simpler and possibly faster than proposal based approaches while avoiding the fundamental flaws mentioned in the intro to this paper at the same time.

**Contact:** me@barvinograd.com

**Slides:**<https://goo.gl/iTC9aS>