/TemplateVersion (2026.1)

www.xueshuxiangzi.com

通过仅标签弹性变形应对隐式标签噪声以释放稳健的语义分割性能

Yechan Kim[1*†], Dongho Yoon[1†], Younkwan Lee[2], Unse Fatima[1], Hong Kook Kim[1], Songjae Lee[3], Sanga Park[3], Jeong Ho Park[3], Seonjong Kang[3], Moongu Jeon[1]

[1]GIST, [2]Samsung Electronics, [3]LIG Nex1

[1] { yechankim, gidong76, unse.fatima } @gm.gist.ac.kr, { hongkook, mgjeon } @gist.ac.kr, [2]youn720.lee@samsung.com, [3] { songjae.lee, sanga.park, jeongho.park1, seonjong.kang } @lignex1.com

Abstract

以往关于图像分割的研究主要集中在处理严重（或显式）的标签噪声，而现实世界的数据集中还存在细微（或隐式）的标签缺陷。这些缺陷源于内在挑战，比如模糊的物体边界和标注者的差异性。虽然不是显而易见的，这种轻微和潜在的噪声仍然可能损害模型的性能。典型的数据增强方法对图像及其标签应用相同的变换，可能会放大这些细微的缺陷、限制模型的泛化能力。在本文中，我们推出了 NSegment+，一种新颖的增强框架，通过将图像和标签的变换解耦，来处理语义分割中的这类现实噪声。通过仅对分割标签引入受控的弹性变形，同时保留原始图像，我们的方法鼓励模型专注于学习稳健的物体结构表示，尽管标签存在轻微的不一致。大量实验表明，NSegment+ 能够持续提高性能，在 Vaihingen、LoveDA、Cityscapes 和 PASCAL VOC 上的平均 mIoU 增益分别达到 +2.29、+2.38、+1.75 和 +3.39——即便没有使用复杂的技巧，仍然突显了解决隐式标签噪声的重要性。当与其他训练技巧（包括 CutMix 和标签平滑）结合使用时，这些增益可以进一步放大。

Code —
https://github.com/unique-chan/NSegmentPlus

介绍

语义分割是计算机视觉中的一项基础任务，涉及为图像的每个像素分配语义标签。尽管在模型架构方面取得了显著进展，分割模型的性能仍然严重依赖于标注数据集的质量 (?) 。然而，创建高质量的像素级标注不仅昂贵且耗时，还容易出现细微的瑕疵。即使是经过精心策划的数据集，通常由于一些固有挑战而包含隐藏的标签噪声（我们称之为“隐性”标签噪声），如不明确的物体边界、混合像素、阴影、遮挡以及标注者之间的不一致 (?) 。与严重和显著的标签噪声（在下文中称为“显性”标签噪声），如缺失的掩码和损坏的类别不同，这些隐性瑕疵即使对于标注专家来说也相对难以检测和纠正，如图 1 所示。然而，即使是个别细微标签噪声的实例看似无害，其分布不一致也可能集体降低模型性能。

最近在噪声标签学习 (LNL) 方面的进展引入了多种策略，包括模型架构的修改、标签优化机制和损失函数设计，以对抗不准确监督的影响 (?) 。虽然这些方法在减轻显性标签错误方面显示了前景，但它们经常依赖于复杂的训练流程，并需要大量的超参数调整。更为重要的是，我们的实验结果表明，现有的 LNL 方法在面对隐性标签噪声时效果不佳，这在先前的工作中常常被忽视。这种隐性噪声虽然不明显，但可能会削弱模型性能。为了解决这一问题，我们寻求另一种方法：一种轻量但高效的解决方案，用于处理现有图像分割基准中的隐性标注噪声。

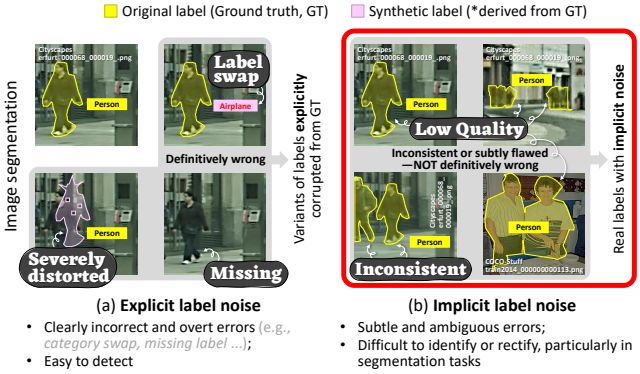

Figure 1: 语义分割中的（a）显性和（b）隐性标签噪声的示例。红色框（□ ）表示本工作中针对的核心挑战。

一种实用的替代方案是扰动训练数据本身，即通过数据增强实现。通过让模型接触到同一输入的多个增强变体，它可以学习关注可泛化的模式，而不是记忆特定的噪声。典型的增强策略 (?) 对图像及其对应标签应用相同的几何变换。问题在于，这种同步增强可能会无意中放大标注噪声，保留甚至加剧标签中的结构不一致性。

为了解决这一挑战，我们提出了 NSegment+，这是一种新的数据增强框架，其将应用于图像和标签的转换解耦。与传统增强方法将标签视为完美的真实值不同，我们的方法承认隐藏的标签不确定性，并利用它来提高模型的鲁棒性。通过保持输入图像不变，仅扭曲分割掩码，我们的方法鼓励模型减少依赖可能有噪声的标签边界，而更多地关注图像中固有的鲁棒语义线索。针对标签特定的变形，我们引入了三个关键创新来增强变形的稳定性和可扩展性：

- 我们首次将弹性变形的应用推广到其在医学图像分割 (?) 中的广泛应用之外，并展示其在一般语义分割任务中的有效性。
- 我们引入了每样本、每时期的随机变形，其中每个分割掩码在每个训练时期中使用随机抽样的“变形幅

*Corresponding authors: Moongu Jeon; Yechan Kim.
†These authors contributed equally.
Preprint.
All rights reserved.

www.xueshuxiangzi.com

度”和“空间平滑度”组合进行独立增强。这个简单但强大的机制给训练过程注入了高变异性，充当了一种标签级别的正则化。

- 我们设计了一种具有尺度意识的形变抑制机制，以保护小物体免受过度变形的影响。在增强过程中，形变场在小标记区域周围被选择性地掩盖。这个组件至关重要，因为对于小掩码的强烈变形可能会导致语义侵蚀。

据我们所知，我们首次定义并解决了语义分割中的隐性标签噪声问题。为了验证 NSegment+ 的广泛适用性，我们在六个不同的基准上进行广泛实验——涵盖遥感（ISPRS Vaihingen、Potsdam、LoveDA）和自然场景（Pascal VOC 2012、Cityscapes、COCO-Stuff 10K）——使用众多最先进的语义分割模型。总体而言，NSegment+ 显示出显著的性能提升，验证了其有效性。我们还展示了 NSegment+ 可以与其他增强和正则化方案有效结合，从而进一步提高性能。

相关工作

计算机视觉中的弹性形变

弹性变形是指一类光滑的、非刚性的变换，这种变换在保持图像连续性和结构完整性的同时改变其形状。最初源于材料科学 (?) ，这一概念已被广泛应用于计算机视觉中，以提高模型的鲁棒性和泛化能力。

例如，(?) 是最早将随机弹性变形应用于手写字符识别训练数据的成果之一。在图像配准和形状分析领域，大变形微分同胚度量映射 (LDDMM) (?) 是一个基于弹性变换原理的著名框架。同样，DeepFlow (?) 结合了类似弹性的变形来优化运动估计。最近，弹性变形在医学图像分析中得到了越来越多的应用，通过在训练期间引入合理的解剖变异，从而增加模型对患者间结构差异的不变性 (?) 。

然而，弹性形变在一般图像分割任务中（例如，地球观测、城市/驾驶场景理解等）很少被采用。这种犹豫主要源于简单的非刚性扭曲往往会将纹理伪影引入输入图像，从而注入不现实的视觉线索，阻碍模型的有效泛化。例如，与 CT 图像不同，RGB 图像表现出显著更高的视觉变异性，因此容易受到纹理误表征的影响。为了弥合这一差距，我们提出了一条专用的增强管道，使弹性形变能够安全有效地集成到通用图像分割中。

在各种计算机视觉任务中，处理标签噪声一直是一个关键挑战。在过去的十年中，学习带有噪声标签（LNL）的问题得到了显著关注，尤其是在图像分类领域。现有的方法大致可以分为三个主要方向：鲁棒架构构建、标签清洗和鲁棒损失函数设计。早期的努力集中于修改模型架构来应对噪声监督。这些方法引入了机制，例如噪声适应层，以吸收被破坏标签的影响。

后续研究转向标签清理策略，包括样本选择 (???) 或标签纠正 (???) 。这些方法通常需要额外的模块，如教师模型或辅助编码器/解码器 (?) 。与此同时，一系列重要的研究工作集中在设计能够容忍标签噪声的稳健损失函数，而不需要进行明确的噪声纠正。例如，引导策略 (??) 和防止错误标签记忆的正则化技术 (???) 。

与此同时，越来越多的工作开始将 LNL 技术从分类扩展到分割，主要是通过在分割流程中引入标签优化机制 (??????) 。

尽管取得了这些进展，大多数现有研究仍然高度依赖于合成噪声设置（见图 1 (a)），这种设置虽然具有控制性和分析上的便利性，但无法反映现实世界噪声分布的复杂性和非结构化特性 (??) 。此外，当前文献主要针对标签污染易于检测的显性噪声。然而，在实际数据集中，标签噪声通常是内隐的，由模糊的物体边界、遮挡或者标注者的差异引起，如图 1 (b) 所示，这些方法未能有效解决这些问题。

大多数 LNL 研究假设同时存在类别噪声和定位噪声。然而，只有少数工作特别将噪声定位作为一个独立挑战进行研究，尽管模型训练——尤其是对空间预测任务的训练——往往对甚至微小的定位错误都极为敏感。在目标检测中，(?) 通过使用三种类型的线性变换（缩放、旋转、平移）显式建模边界框 (BBox) 噪声，开创了这一研究方向，并提出了定位感知损失目标，旨在纠正此类噪声标注。(?) 并没有尝试清除或恢复真实标签，而是通过让模型接触 BBox 注解的多种变体来提高其鲁棒性。该方法仅对定位标签引入各种扰动，而保持对应的图像不变，从而解耦图像和标签的变换。我们将这一研究方向扩展到语义分割，其中隐式定位标签噪声在现有基准的像素空间中进一步呈现空间弥散。

方法论

现实世界的语义分割数据集经常存在隐含的标签噪声。为了克服这一限制，我们设计了一种轻量级但有效的增强机制，命名为 NSegment+ ，它仅在分割标签上操作，以模拟现实世界中的注释模糊。

为了更清晰地理解我们提出的方法，我们首先介绍一种基本形式的标签级变形，我们称之为 NSegment。然后，我们展示我们的最终框架 NSegment+，通过在标签特定变形上引入尺度感知扰动控制来进一步提高鲁棒性。

NSegment：基于标签特定变形的数据增强用于语义分割

本小节介绍我们的核心增强方案 NSegment，它仅对分割标签应用弹性变形。该变换在空间上是平滑的，并在训练时期动态变化，以模拟软边界不确定性和隐含标签噪声。

给定图像 $\mathcal{I} \in \mathbb{R}^{w\times h\times 3}$ 及其对应的分割标签 $\mathcal{L} = \{S_j\}_{j=1}^{C}$ ，其中 $S_j \in \mathbb{R}^{w\times h}$ 表示类别 j 的二值掩码，我们的流程继续如下：

1. 生成随机位移场：为了引入局部非刚性扰动，我们首先生成两个位移场 $d\mathcal{X}$ 和 $d\mathcal{Y}$ ，形状为 $w \times h$ ，如图 2 - ① 所示。每个值从 $[-1, 1]$ 中的均匀分布中抽取。
2. 使用高斯核进行随机平滑：如图 2 - ② 所示，从预定义集合 Ω 中抽取一个随机对 (α, σ) ，其中 α 控制变形的大小（强度），σ 控制空间平滑度（通过二维高斯核实现）。然后，原始位移场 $d\mathcal{X}$ 和 $d\mathcal{Y}$ 中的每个值均按比例缩放 α ，且 $\phi(\cdot)$ 应用零填充以维持原始空间形状。最后，将位移场 $d\mathcal{X}$ 和 $d\mathcal{Y}$ 与高斯核 G_σ 进行卷积。此操作产生平滑且空间一致的变形，使得标签扰动能够模拟柔和且真实的隐式注释噪声，而不会引入尖锐的不连续性。
3. 分割掩码标签的弹性变形：对于每个类别标签 S_j ，应用 x 和 y 轴的平滑位移场 $d\mathcal{X}$ 和 $d\mathcal{Y}$ 重新映射像素坐标。具体来说，使用双线性插值和适当的边界裁剪（算法 1 中的 $\varphi(\cdot)$ ），将 S_j 中的每个像素 (x, y) 移动到 $(x + d\mathcal{X}[x, y], y + d\mathcal{Y}[x, y])$ 。这为每个类别 j 产生了变形后的标签 S_j^* 。

www.xueshuxiangzi.com

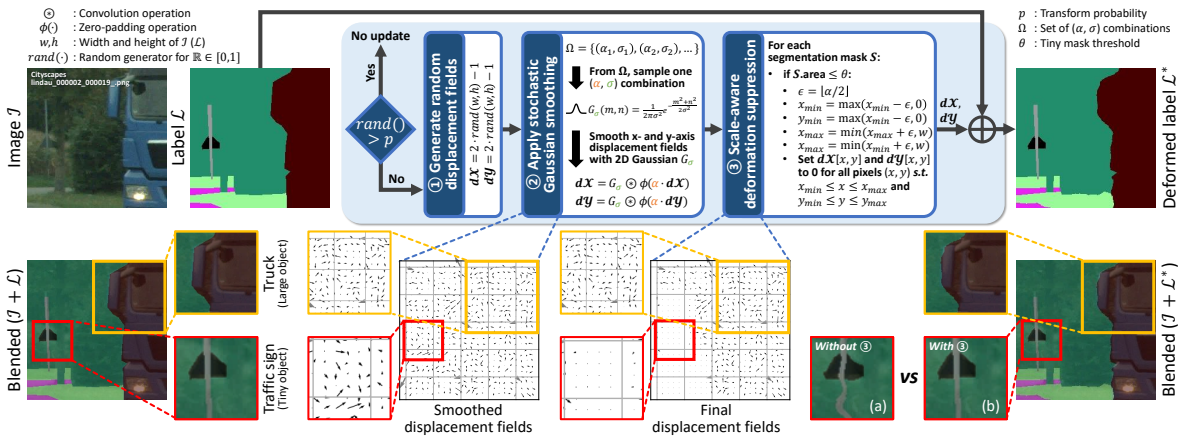

Figure 2: 提出的 NSegment+ 用于语义分割的数据增强的总体流程。为了减轻隐式标签噪声，我们的方法仅对标签掩码进行扰动，同时保持输入图像不变。具体来说，我们引入了两个关键组件：（1）随机高斯平滑以多样化标签，以及（2）一个抑制小掩码失真以保持标签结构完整性的约束。尤其是后者在我们的流程中很重要，以避免如（a）所示的严重语义错位，与（b）相比时更为明显。建议在彩色屏幕上放大查看效果最佳。

注意，上述程序在每一个训练回合中独立应用于每个图像-标签对，仅在 $rand() > p$ 时激活，其中超参数 p 表示变换概率。通过在每次迭代中采样不同的 (α, σ) 组合，模型接触到广泛的几何失真，从而增强了对隐式标签噪声的鲁棒性。此外，这种随机采样消除了参数调优的负担。

NSegment+：将尺度感知扰动策略融入 NSegment

本小节介绍了我们的扩展 NSegment+，通过防止小分割区域的有害变形进一步提高鲁棒性。当 NSegment 对所有标签区域均匀应用变形时，NSegment+ 包含一个尺度感知抑制模块，该模块可选择性地禁止对小掩膜进行变形。该过程通过引入一个每段过滤器来扩展 NSegment：

1. 对于每个区域 S_j，如果其像素面积小于预定的阈值 θ，则会识别出该区域周围的局部邻域（参见算法 1 中的第 14 行）。
2. 在这个邻域内，$d\mathcal{X}$ 和 $d\mathcal{Y}$ 中的相应区域被设为零，有效地保留了小区域的原始标签形状。（详见算法 1 的第 2-6 行或图 2 - ③ ）

这使得小段能够在我们的框架中发生变形，防止语义错位和性能下降，特别是在对象尺度变化较大的数据集中。请参阅算法 1 和图 2 以全面理解我们的框架。

Algorithm 1: NSegment+ 的过程。

Input: Input image $\mathcal{I}$ of size $w \times h$, Its corresponding segmentation labels $\mathcal{L} = \{S_j\}_{j=1}^{C}$, Set of (α, σ) combinations $\mathbf{\Omega} = \{(\alpha_k, \sigma_k)\}_{k=1}^{K}$, Transform probability p , small mask threshold θ

1 function suppresssmallMask ($S_j, d\mathcal{X}, d\mathcal{Y}, \alpha$):
2 Let $x_{min}, y_{min}, x_{max}, y_{max}$ denote the minimum and maximum values of the x and y coordinates among all pixel points in segment S_j ;
3 $\epsilon \leftarrow \lfloor \alpha/2 \rfloor$;
4 $x_{min}, y_{min} \leftarrow max(x_{min} - \epsilon, 0), max(y_{min} - \epsilon, 0)$;
5 $x_{max}, y_{max} \leftarrow min(x_{max} + \epsilon, w), min(y_{max} + \epsilon, h)$;
6 Set $d\mathcal{X}[x,y]$ and $d\mathcal{Y}[x,y]$ to 0 for all pixels (x,y) s.t. $x_{min} \le x \le x_{max}$ and $y_{min} \le y \le y_{max}$;
7 if $rand() > p$ then
8 return $\mathcal{L}$; // No update labels
9 Initialize S_j^* for each category index j as zero-filled matrix of shape $w \times h$;
10 $\alpha, \sigma \leftarrow randChoice(\mathbf{\Omega})$;
11 $d\mathcal{X} \leftarrow G_\sigma \circledast \phi(\alpha(2 \cdot rand(w,h) - 1))$;
12 $d\mathcal{Y} \leftarrow G_\sigma \circledast \phi(\alpha(2 \cdot rand(w,h) - 1))$;
13 for $j \leftarrow 1$ to C do
14 if $S_j.area \le \theta$ then
15 suppresssmallMask ($S_j, d\mathcal{X}, d\mathcal{Y}, \alpha$);
16 Set $S_j^*[\varphi(x + d\mathcal{X}[x,y]), \varphi(y + d\mathcal{Y}[x,y])]$ to $S_j[x,y]$ for each pixel (x,y) in segment S_j ;
17 $\mathcal{L}^* \leftarrow (S_j^*)_{j=1}^{C}$;
18 return $\mathcal{L}^*$; // Update labels

实验与分析

实验装置

为了评估我们提出的框架的有效性和泛化能力，我们在六个不同的语义分割基准上进行了广泛的实验：Vaihingen (?) 、Potsdam (?) 、LoveDA (?) 、Cityscapes (?) 、Pascal VOC 2012 (?) 和 COCO-Stuff 10K (?) 。这些数据集包括遥感影像（Vaihingen、Potsdam、LoveDA）和自然场景（Cityscapes、Pascal VOC、COCO-Stuff），从而能够全面评估在不同领域的表现。在所有实验中，我们使用平均交并比（mIoU）作为评估指标。为确保统计的可靠性，每个实验使用不同的随机种子进行五次，我们报告这五次运行的 mIoU 值的平均值和标准差。

我们使用基于 PyTorch 的 MMSegmentation 框架实现了我们所有的模型。我们的方法被实现为一个自定义变换模块，并集成到标准的数据预处理流程中。为了确保公平的比较，我们遵循每个数据集特定的常规训练和评估协议。我们在广泛的最新分割模型上评估我们方法的有效性。我们自动为每个 σ 设置高斯核大小，使用与 OpenCV 标准一致的公式 $2 \cdot \lfloor 3\sigma \rceil + 1$ 。所有数据集中的掩膜变形抑制阈值 θ 设置为 1000。变换概率 p

www.xueshuxiangzi.com

Table 1: NSegment 和 NSegment+ 对遥感场景中训练最先进的分割模型的影响

Datasets	Vaihingen			Potsdam			LoveDA		
Models	Baseline	NSegment	NSegment+	Baseline	NSegment	NSegment+	Baseline	NSegment	NSegment+
DeepLab V3+ (ECCV 18)	77.53 ± 0.30	78.26 ± 0.63 (+0.73)	78.34 ± 0.26 (+0.81)	82.95 ± 0.10	83.16 ± 0.11 (+0.21)	83.20 ± 0.03 (+0.25)	43.29 ± 0.77	43.36 ± 0.62 (+0.07)	43.60 ± 0.17 (+0.31)
ANN (ICCV 19)	79.75 ± 0.07	80.44 ± 0.12 (+0.69)	80.60 ± 0.16 (+0.85)	84.81 ± 0.06	84.84 ± 0.03 (+0.03)	84.88 ± 0.13 (+0.07)	45.93 ± 0.69	47.22 ± 0.19 (+1.29)	47.32 ± 0.58 (+1.39)
DANet (CVPR 19)	79.59 ± 0.61	79.71 ± 0.24 (+0.12)	79.93 ± 0.40 (+0.34)	84.47 ± 0.24	84.51 ± 0.23 (+0.04)	84.67 ± 0.14 (+0.20)	43.35 ± 0.78	43.82 ± 0.89 (+0.47)	44.16 ± 0.61 (+0.81)
APCNet (CVPR 19)	79.15 ± 0.85	80.39 ± 0.30 (+1.24)	80.49 ± 0.35 (+1.34)	84.73 ± 0.13	84.74 ± 0.06 (+0.01)	84.88 ± 0.01 (+0.15)	46.60 ± 0.42	46.79 ± 1.06 (+0.19)	47.25 ± 1.12 (+0.65)
GCNet (TPAMI 20)	79.60 ± 0.78	80.55 ± 0.36 (+0.95)	80.38 ± 0.05 (+0.78)	84.90 ± 0.12	84.91 ± 0.17 (+0.01)	84.92 ± 0.12 (+0.02)	46.46 ± 0.60	46.65 ± 0.63 (+0.19)	46.70 ± 1.34 (+0.24)
OCRNet (ECCV 20)	75.39 ± 0.79	76.80 ± 0.42 (+1.41)	77.68 ± 0.36 (+2.29)	82.62 ± 0.11	82.63 ± 0.22 (+0.01)	82.71 ± 0.11 (+0.09)	44.26 ± 0.66	44.28 ± 0.04 (+0.02)	44.61 ± 0.08 (+0.35)
Mask2Former (CVPR 22)	77.47 ± 0.22	77.85 ± 0.11 (+0.38)	77.93 ± 0.09 (+0.46)	82.73 ± 0.39	83.06 ± 0.07 (+0.33)	83.20 ± 0.46 (+0.47)	48.50 ± 0.68	48.82 ± 0.37 (+0.32)	48.84 ± 0.40 (+0.34)
DOCNet (GRSL 23)	80.80 ± 0.32	81.41 ± 0.15 (+0.61)	81.42 ± 0.06 (+0.62)	85.61 ± 0.05	85.69 ± 0.05 (+0.08)	85.70 ± 0.03 (+0.09)	50.92 ± 0.50	51.12 ± 0.42 (+0.20)	51.40 ± 0.45 (+0.48)
CAT-Seg (CVPR 24)	71.59 ± 0.20	71.62 ± 0.09 (+0.03)	71.66 ± 0.10 (+0.07)	82.49 ± 0.16	82.58 ± 0.04 (+0.09)	82.61 ± 0.04 (+0.12)	46.89 ± 0.58	47.24 ± 0.39 (+0.35)	47.46 ± 0.28 (+0.57)
Golden (CVPR 25)	70.67 ± 0.34	71.34 ± 0.53 (+0.67)	71.62 ± 0.50 (+0.95)	78.31 ± 0.26	78.51 ± 0.11 (+0.20)	78.76 ± 0.10 (+0.45)	37.56 ± 1.91	39.16 ± 0.92 (+1.60)	39.94 ± 0.99 (+2.38)
LOGCAN++ (TGRS 25)	80.97 ± 0.08	81.04 ± 0.14 (+0.07)	81.09 ± 0.06 (+0.12)	86.07 ± 0.06	86.11 ± 0.14 (+0.04)	86.12 ± 0.01 (+0.05)	50.59 ± 0.13	50.77 ± 0.34 (+0.18)	51.23 ± 0.56 (+0.64)

Note: For each data-model combination, the highest result is highlighted in bold whereas the second-best is underlined in this paper

Table 2: 不同 α - σ 组合对 NSegment 中固定和随机标签级别弹性变形的影响

Baseline	Fixed α - σ						Stochastic α - σ sampling
	σ \ α	1	5	30	50	100	
77.41	3	77.57	77.50	77.51	77.25	75.99	77.75 (+0.34)
	5	77.30	77.70	77.41	77.24	77.54	
	10	76.49	77.34	77.67	77.55	77.66	

Table 3: 在 NSegment 中，图像 vs. 仅标签弹性变形对语义分割的影响

For all cases, we adopt stochastic α - σ sampling	Elastic deformation		Test mIoU
	Image	Label	
Baseline	-	-	77.41
+ (a) Normal (Identical image-label transform)	✓	✓	67.58 (-9.83)
+ (b) Transform only for images	✓		70.69 (-6.72)
+ (c) Transform only for labels		✓	77.75 (+0.34)

默认为 0.5，但在表格 2 和 3 中设置为 1.0。此外，我们用 $\mathbf{\Omega} = (\alpha_k, \sigma_k)_k$ 表示根据网格搜索基于两个集合 $1, 15, 30, 50, 100$ 和 $3, 5, 10$ 的笛卡尔积。数据和模型特定的训练配置（学习率、批量大小、图像预处理等）记录在补充材料中。

与集中于明显（或显性）标注错误的典型研究相比，我们做了一套不同的假设：（1）所有数据集都没有显性标签错误，（2）因为真实标签是不可观测的，有时也是模糊的，所以每个数据本质上都受到隐性标签噪声的影响。因此，我们在训练和评估中都采用它们的原始形式。每个数据集可能内在地包含不同且不可量化形式的隐性标签噪声。因此，我们的方法在这些异质数据集中，始终优于基线模型（即未经 NSegment+ 训练的模型），提供了其在真实世界场景中稳健性和实际相关性的有力证据，因为无法保证绝对干净的标注。

消融研究

我们进行了全面的消融研究，以验证在我们的 NSegment+ 框架中每个组件的有效性。具体来说，我们分析了 (1) 通过随机高斯平滑实现的多样化变形强度的好处，(2) 不同变形目标的影响，和 (3) 小掩码变形抑制的作用。针对 (1) 和 (2)，我们采用 PSPNet (?) 并在 Vaihingen (?) 数据集上进行实验。

1) **随机高斯平滑的好处** 表 2 显示，对于仅标签的变形，随机采样变形参数 (α, σ) 在整体上优于固定配置。虽然固定设置可以相较于基线带来微小的改进，但对于

Table 4: 在没有 NSegment 和 NSegment+ 的基线上的平均性能提升。NSegment+ 始终获得更大的增益，突出了尺度感知变形抑制的优势。

	Scale-aware deformation suppression	Vaihingen	Potsdam	LoveDA	City-scapes	PASCAL VOC 2012	COCO-Stuff 10K
NSegment		0.63	0.10	0.44	0.41	0.57	0.19
NSegment+	✓	0.78 (+0.15)	0.18 (+0.08)	0.74 (+0.30)	0.83 (+0.42)	0.91 (+0.34)	0.44 (+0.25)

(α, σ) 的最佳选择仍不明确。例如，$(\alpha = 5, \sigma = 5)$ 在 Vaihingen 上效果尚可，但可能无法推广到其他数据集或模型。与此相反，我们的随机平滑策略在每次迭代时从一个广泛的参数范围 $\mathbf{\Omega}$ 随机采样 (α, σ)。它带来了两个主要好处：i）通过让模型接触多样的标签变形（一种标签级别的正则化），持续提升性能；ii）消除了手动参数调整的需要，前提是参数空间 $\mathbf{\Omega}$ 配置得当。

2) **不同变形目标的影响** 表格 3 显示，对图像和标签同时应用相同的弹性变换会严重降低性能（-9.83），这突出了在模型训练中不切实际的外观失真。类似的现象可以在仅对图像进行弹性变换时观察到（-6.72）。相反，只对标签应用变形不仅避免了视觉损坏，还提高了 mIoU 值 +0.34。

3) **小掩模变形抑制的作用** 表 4 展示了尺度感知变形抑制模块的好处，该模块在标签变形过程中有选择地禁用了对小物体的扭曲。此表报告了每个数据集相对于基线的平均性能增益（以 mIoU 表示），取平均值于表 1 和 5 中列出的所有分割模型。例如，在 Cityscapes 数据集中，NSegment 的 mIoU 提高了 +0.41，而 NSegment+ 达到了 +0.83，由于小遮罩抑制，产生了 +0.42 的绝对变化。这种模式在所有数据集中是一致的。这清楚地表明，简单地对所有区域进行变形——不考虑物体的尺度——可能会降低性能，尤其是在存在许多小实例或过度分割的遮罩的情况下。

我们评估了 NSegment 和 NSegment+ 在各种最先进的语义分割模型和数据集上的影响。具体而言，我们采用了 12 种分割模型，包括 DeepLab V3+ (?) 、ANN (?) 、DANet (?) 、APCNet (?) 、GCNet (?) 、OCRNet (?) 、SegFormer (?) 、Mask2Former (?) 、DOCNet (?) 、CAT-Seg (?) 、Golden (?) 和 LOGCAN++ (?) 。这些模型涵盖了从 2018 到 2025 年的基于卷积神经网络

www.xueshuxiangzi.com

Table 5: NSegment 和 NSegment+ 对自然场景中训练最先进分割模型的影响

Datasets	Cityscapes			PASCAL VOC 2012			COCO-Stuff 10K		
Models	Baseline	NSegment	NSegment+	Baseline	NSegment	NSegment+	Baseline	NSegment	NSegment+
DeepLab V3+ (ECCV 18)	56.95 ± 0.91	57.41 ± 0.70 (+0.46)	57.58 ± 1.08 (+0.63)	61.64 ± 0.53	61.80 ± 0.55 (+0.16)	61.84 ± 0.67 (+0.20)	21.87 ± 0.11	21.94 ± 0.31 (+0.07)	21.96 ± 0.49 (+0.09)
ANN (ICCV 19)	62.20 ± 0.87	62.33 ± 0.64 (+0.13)	63.33 ± 1.09 (+1.13)	67.87 ± 0.65	68.53 ± 0.22 (+0.66)	68.66 ± 0.98 (+0.79)	29.50 ± 0.25	29.51 ± 0.25 (+0.01)	29.63 ± 0.17 (+0.13)
DANet (CVPR 19)	62.91 ± 0.38	64.21 ± 0.06 (+1.30)	64.66 ± 0.96 (+1.75)	63.52 ± 1.47	63.94 ± 1.93 (+0.42)	64.36 ± 1.72 (+0.84)	27.94 ± 0.26	28.02 ± 0.62 (+0.08)	28.13 ± 0.42 (+0.19)
APCNet (CVPR 19)	61.97 ± 1.17	62.71 ± 0.14 (+0.74)	63.24 ± 0.59 (+1.27)	63.04 ± 0.64	63.45 ± 1.32 (+0.41)	63.82 ± 0.36 (+0.78)	27.52 ± 0.44	27.86 ± 0.33 (+0.34)	27.94 ± 0.36 (+0.42)
GCNet (TPAMI 20)	62.44 ± 0.34	62.66 ± 0.52 (+0.22)	62.71 ± 0.11 (+0.27)	61.07 ± 3.17	63.50 ± 1.17 (+2.43)	64.46 ± 2.74 (+3.39)	23.86 ± 0.54	24.32 ± 0.11 (+0.46)	24.45 ± 0.47 (+0.59)
OCRNet (ECCV 20)	55.94 ± 1.71	56.11 ± 1.52 (+0.17)	56.51 ± 1.21 (+0.57)	59.65 ± 1.39	59.78 ± 1.25 (+0.13)	60.23 ± 0.86 (+0.58)	22.23 ± 0.51	22.75 ± 0.30 (+0.52)	22.84 ± 0.28 (+0.61)
SegFormer (NeurIPS 21)	62.69 ± 0.14	62.71 ± 0.08 (+0.02)	62.84 ± 0.52 (+0.15)	69.70 ± 0.37	69.85 ± 0.35 (+0.15)	70.00 ± 0.42 (+0.30)	29.07 ± 0.46	29.16 ± 0.26 (+0.09)	29.37 ± 0.06 (+0.30)
Mask2Former (CVPR 22)	62.98 ± 1.12	63.50 ± 0.36 (+0.52)	63.72 ± 0.54 (+0.74)	71.02 ± 0.70	71.14 ± 0.40 (+0.12)	71.68 ± 0.38 (+0.66)	32.84 ± 0.31	32.93 ± 0.35 (+0.09)	34.13 ± 0.32 (+1.29)
Golden (CVPR 25)	74.09 ± 0.50	74.21 ± 0.65 (+0.12)	75.06 ± 0.55 (+0.97)	41.33 ± 1.07	41.95 ± 0.58 (+0.62)	42.02 ± 0.20 (+0.69)	10.92 ± 0.39	10.93 ± 0.22 (+0.01)	11.28 ± 0.22 (+0.36)

Table 6: 结合 NSegment+ 和现有方法的影响，包括图像级增强和对数级正则化策略在六个不同的语义分割基准上

Dataset	Method	No augmentation/ regularization used	Horizontal Flipping (HF)	Random Resize (RR)	Photometric Distortion (PD)	CutOut (CO)	CutMix (CM)	Random Erasing (RE)	Label Smoothing (LS)
Vaihingen	Baseline	77.53 ± 0.30	78.62 ± 0.19	80.04 ± 0.38	77.33 ± 0.35	78.29 ± 0.69	77.54 ± 0.60	79.01 ± 0.27	77.97 ± 0.28
	w/ ours	78.34 ± 0.26 (+0.81)	78.95 ± 0.23 (+0.33)	80.38 ± 0.13 (+0.34)	77.68 ± 0.32 (+0.35)	78.80 ± 0.33 (+0.51)	78.34 ± 0.27 (+0.80)	79.06 ± 0.20 (+0.05)	78.47 ± 0.14 (+0.50)
Potsdam	Baseline	82.95 ± 0.10	84.00 ± 0.12	84.37 ± 0.13	82.76 ± 0.13	83.27 ± 0.03	82.99 ± 0.05	83.43 ± 0.04	82.84 ± 0.13
	w/ ours	83.20 ± 0.03 (+0.25)	84.12 ± 0.07 (+0.12)	84.44 ± 0.17 (+0.07)	83.00 ± 0.03 (+0.24)	83.31 ± 0.14 (+0.04)	83.08 ± 0.05 (+0.09)	83.53 ± 0.01 (+0.10)	83.03 ± 0.19 (+0.19)
LoveDA	Baseline	43.29 ± 0.77	43.48 ± 0.22	40.73 ± 1.05	41.86 ± 1.51	43.40 ± 0.23	43.21 ± 0.83	41.84 ± 0.50	43.05 ± 0.47
	w/ ours	43.60 ± 0.17 (+0.31)	43.92 ± 0.24 (+0.44)	45.89 ± 0.40 (+5.16)	43.10 ± 0.39 (+1.24)	43.91 ± 0.49 (+0.51)	44.18 ± 0.53 (+0.97)	42.19 ± 0.43 (+0.35)	43.64 ± 0.25 (+0.59)
Cityscapes	Baseline	56.95 ± 0.91	56.62 ± 1.24	72.95 ± 0.27	54.10 ± 0.49	55.92 ± 0.15	56.28 ± 0.96	56.28 ± 0.16	55.64 ± 0.42
	w/ ours	57.58 ± 1.08 (+0.63)	58.35 ± 0.47 (+1.73)	73.13 ± 0.44 (+0.18)	54.60 ± 0.39 (+0.50)	57.15 ± 0.68 (+1.23)	56.92 ± 0.50 (+0.64)	56.43 ± 0.99 (+0.15)	56.56 ± 0.83 (+0.92)
PASCAL VOC 2012	Baseline	61.64 ± 0.53	61.80 ± 0.59	55.95 ± 0.80	61.30 ± 0.23	61.56 ± 0.35	61.59 ± 0.86	61.03 ± 0.88	62.55 ± 0.72
	w/ ours	61.84 ± 0.67 (+0.20)	61.85 ± 0.39 (+0.05)	56.94 ± 0.68 (+0.99)	61.55 ± 0.51 (+0.25)	61.61 ± 0.60 (+0.05)	61.97 ± 0.36 (+0.38)	61.37 ± 0.37 (+0.34)	62.90 ± 0.44 (+0.35)
COCO-Stuff 10K	Baseline	21.87 ± 0.11	21.67 ± 0.48	18.90 ± 0.25	21.37 ± 0.09	21.82 ± 0.11	21.81 ± 0.28	21.37 ± 0.31	21.73 ± 0.33
	w/ ours	21.96 ± 0.49 (+0.09)	21.82 ± 0.36 (+0.15)	19.08 ± 0.38 (+0.18)	21.60 ± 0.19 (+0.23)	21.85 ± 0.21 (+0.03)	21.94 ± 0.32 (+0.13)	21.55 ± 0.36 (+0.18)	22.00 ± 0.13 (+0.27)

Table 7: 将 NSegment+ 与现有的为处理显式标签噪声而设计的 LNL 方法进行比较

	Vaihingen	Cityscapes	GFLOPs
Baseline	77.53 ± 0.30	56.95 ± 0.91	54.27 (1x)
Compensation Learning	77.68 ± 0.30 (+0.15)	57.11 ± 0.79 (+0.16)	55.04 (1.01x)
L2B	77.72 ± 0.43 (+0.19)	56.51 ± 1.06 (-0.44)	54.27 (1x)
UCE	77.91 ± 0.12 (+0.38)	57.01 ± 1.57 (+0.06)	542.69 (10x)
AIO2	77.73 ± 0.27 (+0.20)	57.49 ± 1.14 (+0.54)	108.54 (2x)
NSegment+	78.34 ± 0.26 (+0.81)	57.58 ± 1.08 (+0.63)	54.27 (1x)

和 Transformer 的架构。在六个不同的基准上进行实验，包括遥感数据集（Vaihingen, Potsdam, LoveDA）和自然场景数据集（Cityscapes, VOC, COCO-Stuff），涵盖了一系列输入模式和注释特征。表格 1 和 5 总结了每个模型在有无我们提出的增强方法情况下的 mIoU 表现。NSegment 稳定地提升了相对于原始基线的性能，而 NSegment+ 通过保持小物体的结构完整性进一步提升了性能。例如，在 PASCAL VOC 上，NSegment+ 相比基线促进了 GCNet 的 mIoU 提升了 +3.39，而 NSegment 则是提升了 +2.43。

NSegment+ 与现有 LNL 方法的比较

表格 7 将 NSegment+ 与最新的噪声标签学习（LNL）方法进行基准测试，包括补偿学习 (?)、L2B (?)、UCE (?) 和 AIO2 (?)，在 Vaihingen 和 Cityscapes 上使用 DeepLab V3+ 进行对比。与这些 LNL 方法相比，这些方法通常依赖于不确定性建模、伪标签细化或辅助网络，而 NSegment+ 更简单且与架构无关。此外，与需要额外 10 × 和 2 × FLOPs 的 UCE 和 AIO2 不同，NSegment+ 没有额外的计算成本并实现了最佳性能。这些结果强调了隐式标签噪声尽管在先前的 LNL 文献中常被忽视，仍值得认真考虑。尽管 NSegment+ 独立表现优越，但我们认为将其与现有的 LNL 策略结合可能会提供互补优势，特别是在同时存在隐式和显式标签噪声的情况下。对此的深入分析不在本文的讨论范围内，将留待未来工作进行。

将 NSegment+ 与现有的数据增强或对数级别的正则化策略相结合

我们研究了 NSegment+ 是否能补充现有的增强技术（例如，CutOut (?)，CutMix (?)，随机擦除 (?)）及 logit 层面的正则化（例如，标签平滑 (?)），在六个基准数据集上，与 DeepLab V3+ 结合使用。表 6 展示了当 NSegment+ 与每种策略结合使用时，能够稳定地提高 mIoU。例如，在 LoveDA 数据集上，随机大小调节与我们的组合比单独使用随机大小调节提高了 +5.16 mIoU。这是在 LoveDA 上测试的所有变种中提升幅度最大的。强大的协同效应可能是因为随机大小调节在图像层面上多样化了输入尺度，而 NSegment+ 在标签层面引入了隐式的边界噪声——共同推动模型在具有挑战性的城乡变化中学习尺度不变和标签容忍的特征。

此外，在 PASCAL VOC 上，将 Label Smoothing (?) 与 NSegment+ 结合使用可获得最佳性能，优于所有其他组合（62.90 mIoU）。Label Smoothing 可以防止在难以分类的类别上出现过度自信，而 NSegment+ 则鼓励对空间不精确性的鲁棒性——两者共同在这两个维度上形成互补的正则化效果。

www.xueshuxiangzi.com