使用完全卷积双流融合网络进行交互式图像分割

A fully convolutional two-stream fusion network for interactive image segmentation

Yang Hu[a,*], Andrea Soltoggio [a], Russell Lock [a], Steve Carter

摘要

在本文中，我们提出了一种新的完全卷积双流融合网络（FCTSFN）用于交互式图像分割。所提出的网络包括两个子网络：双流后期融合网络（TSLFN）对已经降低分辨率的图像前景进行预测，以及一个多尺度精炼网络（MSRN）在全分辨率下提取前景。 TSLFN 包括两个不同的深层流和融合网络。直观上，因为用户交互过程中，对图像的前景和背景的关注超过图像本身，TSLFN 的双流结构减少了用户直接操作和网络输出之间的层数，使用户指定的特性对分割结果产生更直接的影响。MSRN 融合了具有不同的尺度不同层次的 TSLFN，以便在前景上寻找以全分辨率细化的从局部到全局的信息分割结果。我们基于四个基准数据集进行了全面的实验。结果表明，所提出的网络性能与当前最先进的交互式图像分割方法可以媲美。

1. 简介

二值图像分割旨在将图像分离为，关注对象（前景）和其他部分（背景）。 它具有广泛的应用，例如医学影像分析，图像编辑，对象检索等。但是，由于关注对象在不同的环境中变化很大，大多数全自动的方法仅能针对特定的对象进行定制和优化在某种应用中。 很难开发出适用于一般情况的全自动方法。

为了提高图像分割的灵活性和通用性，许多算法采用交互式框架。这些算法允许用户与一个系统交互通过对前景背景像素进行标记以指定兴趣对象。最传统的交互式图像分割算法（Bai & Sapiro, 2007; Boykov & Jolly, 2001; Grady, 2006;Gulshan, Rother, Criminisi, Blake, & Zisserman, 2010; Price, Morse,& Cohen, 2010; Rother, Kolmogorov, & Blake, 2004; Vezhnevets& Konouchine, 2005）依靠低级特征来估计从用户标记的像素到的前景/背景分布预测未标记像素的类别。这些方法存在的问题是低级特征可能无法在许多情况下区分前景和背景，例如，前景和背景具有相似的颜色和质地; 或者前景包括几个部分和不同的外观。 因此，基于低级特征的算法可能需要大量的用户交互才能获得可靠的细分，最终增加了用户的负担。

最近，由深度神经网络（DNNs）发掘的深层特征已经在许多计算机视觉任务中展示了它们的力量包括图像分类(He, Zhang, Ren, & Sun, 2016;Krizhevsky, Sutskever, & Hinton, 2012; Simonyan & Zisserman,2015)图像分割(Chen, Papandreou, Kokkinos,Murphy, & Yuille, 2017; Li, Qi, Dai, Ji, & Wei, 2017; Shelhamer,Long, & Darrell, 2017; Zheng et al., 2015)。因此许多研究人员(Feng, 2017; Mahadevan, Voigtlaender, & Leibe, 2018; Maninis,Caelles, Pont-Tuset, & Van Gool, 2018; Wang et al., 2017; Xu, Price,Cohen, Yang, & Huang, 2016, 2017) 已经使用对图像和用户指定特征有更深层次的理解 DNN 网络提取深层特征，以改善交互式图像分割。 大多数这些基于 DNN 的方法可以看作是早期的特征融合 DNN。它们将关联的图像和关于图像特征的用户交互作为 DNN 的输入; 通常， DNN 被用作结合相关特性预测前景和背景。 然而，这种早期融合方案可能不能充分利用用户指定的信息来预测前景/背景。 具体来说，考虑到最先进的技术 DNN 通常由大量层组成，在早期融合用户操作与图像特征可能会削弱用户对最终预测结果的影

响。

与现有的早期融合网络相比，我们认为通过后期融合结构可以得到更好的表现，使用两个单独的深流来学习和提取图像的深层特征和用户交互，然后融合来自两个流的特征。 我们的直觉是这种后期融合结构允许用户指定特征对预测结果产生更直接的影响，因为用户标注特性和预测结果输出之间的层数较少。我们预计这会带来性能改善，因为用户指定的前景/背景的位置是相比图像本身更直接的信息。在同时，两个独立的流仍然产生了深层特征，所以整个网络仍然保留深层特征的代表性优势。这允许网络准确理解图像内容并预测兴趣对象。

在本文中，我们提出了一种新颖的完全卷积双流融合网络（FCTSFN）用于交互式图像分割。如图 1 所示，我们提出的网络以末期融合双流网络（TSLFN）。TSLFN 分别独立使用两个独立的流提取来自图像的深层信息和用户指定的信息，同时使用融合网络融合从两个流产生的特征来预测前景和背景。所以这种双流末期融合结构减少了用户指定特征和网络输出之间的层数，我们期望它能够改善用户交互对预测结果的影响，实现更好的分割性能。此外，为了处理 TSLFN 中的分辨率损失，我们使用多尺度精炼网络（MSRN）来细化 TSLFN 的结果至全分辨率。 MSRN 融合了 TSLFN 产生的不同尺度不同层次的特征。预期融合结果应当包括用户指定对象上局部到全局结构信息，并且 MSRN 可以利用融合结果来细化 TSLFN 输出至全分辨率。本文的贡献如下

如下：

- 我们提出了一种新颖的完全卷积双流融合网络（FCTSFN）用于交互式图像分割。
- 在 FCTSFN 中，我们提出了一种双流末期融合网络（TSLFN）旨在改善用户交互对预测结果的影响以实现更好的分割性能。
- 在 FCTSFN 中，我们提出了一个多尺度的精炼网络（MSRN）融合不同尺度的信息

细化 TSLFN 的输出。

在本文的其余部分编排如下。第 2 节，介绍相关的成果。第 3 节，详细介绍我们提出的用于交互式图像分割的 FCTSFN。第 4 节，分析和比较的实验结果。第 5 节，总结。

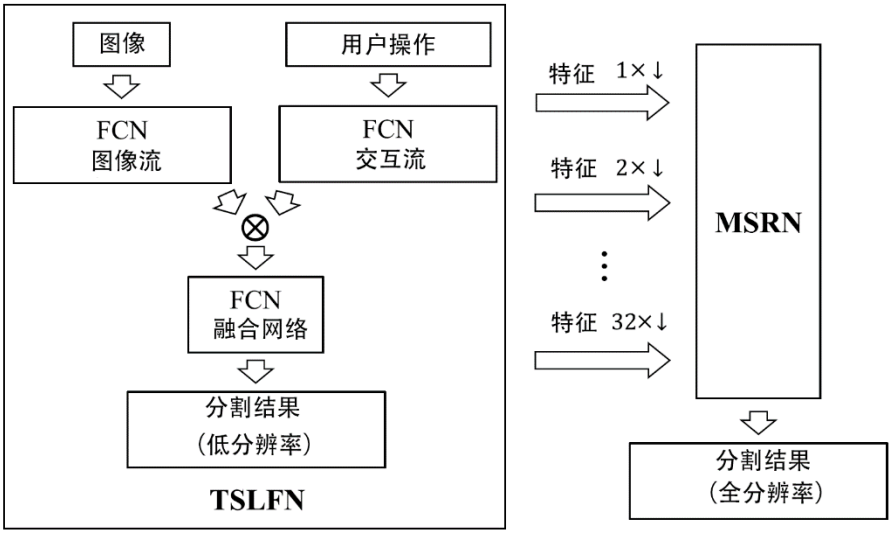


图 1 网络架构的流程图。⊗为级联操作; n× ↓为 n 次下采样。

2. 相关成果

目前已经有大量的交互式图像分割方法提出。Boykov and Jolly (2001)提出基于图形切割方法。此方法将图像表示为图形，像素被视为图节点，相邻节点被认为是由边连接。 有了这个图形结构，交互式图像分割被公式化为能量最小化问题这可以通过图形切割来解决(Boykov,

Veksler, & Zabih, 2001; Kolmogorov & Zabin, 2004). 继 Boykov 和 Jolly 之后 Rother et al. (2004)等，提出了 GrabCut，对图形进行迭代地削减。使用用户提供的初始边界，GrabCut 在前景/背景分布估计和图形切割分割之间进行迭代，以逐步细化前景和背景。利用与 Boykov 和 Jolly（2001）类似的图形表示，Bai 和 Sapiro（2007）使用未标记像素和用户标记的前景/背景像素之间的测地距离来确定前景和背景。为了利用图形切割和测地距离，Price 等人（2010）。提出了测地图切割，其将测地距离结合到基于图切割的框架中。 此外，Gulshan 等人。（2010）通过将具有星形凸度的测地距离作为分割结果的形状约束来改进图形切割方法（Boykov＆Jolly，2001）。在其他有代表性的研究中，Vezhnevets 和 Konouchine（2005）提出了 GrowCut，它基于细胞自动机迭代地更新像素标签（Neumann，1966）。 Grady（2006）提出了一种随机游走方法。该方法计算从每个未标记的像素开始随机游走者首先到达用户标记的像素的概率； 然后，基于具有最高概率的用户标记像素来分配像素标签。 所有上述方法都利用低级特征（颜色，纹理等）来模拟前景和背景分布。因此，它们区分前景和背景的性能受到低级特征适用性的限制。因此，对于难以使用低级特征区分前景和背景差异的复杂场景，这些方法可能需要用户标记大量像素以实现良好的分割结果。这增加了用户的负担。最近，深度神经网络（DNN）在区分图像中的不同对象方面表现出优越的性能(Chen,Papandreou et al., 2017; Li et al., 2017; Shelhamer et al., 2017;Zheng et al., 2015)。此外，从 DNNs 习得的深层特征被证明可以很好的转移到其他问题(Oquab, Bottou, Laptev, & Sivic, 2014; Zeiler & Fergus, 2014)。因此，一些研究人员专注于应用 DNN 来获得对图像和用户交互具有更高层次理解的特征，以改善交互式图像分割。Xu 等人（2016）。使用两个欧几里德距离图来表示用户的正面和负面点击。它们通过将两个距离图与图像的 RGB 通道连接来形成图像-用户交互对。训练完全卷积网络（FCN）从图像-用户交互对预测前景/背景。Boroujerdi 等人（2017）使用类似的图像-用户交互对作为网络的输入。使用完全卷积网络来预测前景/背景。这个网络取代了 Xu 等人（2016）在 FCN 中的最后两个卷积层，通过具有的三个卷积层，内核尺寸逐渐减小，能够更好地捕获对象的几何形状。Wang 等人。 （2017）将用户交互转换为两个测地距离图。他们构建了与 Xu 等人类似的图像 - 用户交互对。（2016）但通过他们通过额外的 DNN 生成分割方案。他们使用保留分辨率网络预测前景/背景。 Xu 等人。（2017）提出 Deep GrabCut。该方法可以从用户提供的边界框中寻找对象边界。它将边界框转移到距离图中。编码- 解码网络用于从级联图像和距离图预测前景/背景。Maninis 等。（2018）从极端点寻求前景。他们将极值点编码为 2D 高斯信号，并与输入图像连接; 残差网络（He et al.，2016）和金字塔场景解析模型（Zhao，Shi，Qi，Wang，＆Jia，2017）用于预测前景。Li，Chen 和 Koltun（2018）使用分段网络从图像和用户交互中生成各种潜在的分段； 然后应用选择网络来选择潜在分段的输出。Mahadevan 等人。（2018）提出了迭代训练算法。该算法不是使用固定的用户点击进行训练，而是根据网络预测的错误逐步增加点击次数。该算法可以提高性能，因为它与真实用户的模式更紧密地对齐。 基本上，所有这些网络（Boroujerdi 等，2017; Li 等，2018; Mahadevan 等，2018; Maninis 等，2018; Wang 等，2017; Xu 等，2016,2017） 采用早期融合结构。它们结合了 DNN 第一层的图像和用户交互功能。与它们不同，本文所提出的 FCTSFN 分别单独提取深度图像和用户交互功能，然后融合它们。与我们提出的网络最相似的工作是 Liew 等人的工作。（2017 年）。 这也是一个双分支网络：它包括一个产生粗略全局预测的全局分支和一个利用多尺度空间金字塔特征进行精细局部预测的局部分支; 最终预测是两个分支的综合结果。然而，所提出的网络与 Liew 等人的网络在三个重要方面不同。（2017）。 首先，Liew 等人。 将图像和交互图连接起来作为网络的输入； 所提出的网络使用两个单独的流来从图像和交互图中提取特征，以允许用户交互对分割结果具有更直接的影响。第二，Liew 等。利用尖端端金字塔池在末端网络来产生多尺度特征； 所提出的网络利用并

融合来自网络的不同层的特征，以将诸如颜色和边缘的低级信息和更高级别的对象信息结合到前景预测中。第三，Liew 等。 使用多尺度特征来细化局部分割，然后将其与全局分割相结合;所提出的网络凭借多尺度特征使用直接全局预测细化结构来以全分辨率进行预测。

3.本文提出的网络

在本节中，我们提出了用于交互式图像分割的完全卷积双流融合网络（FCTSFN）。 首先，我们描述了整个 FCTSFN 架构中的两个流晚期融合网络（TSLFN）的架构。 然后，我们在 FCTSFN 中呈现多尺度精炼网络（MSRN）的结构。 接下来，我们将演示网络训练流程。最后，我们描述了整个 FCTSFN 的数据处理过程，包括从用户交互生成用户交互图作为网络输入的方法，以及根据网络输出生成前景掩模的方法。

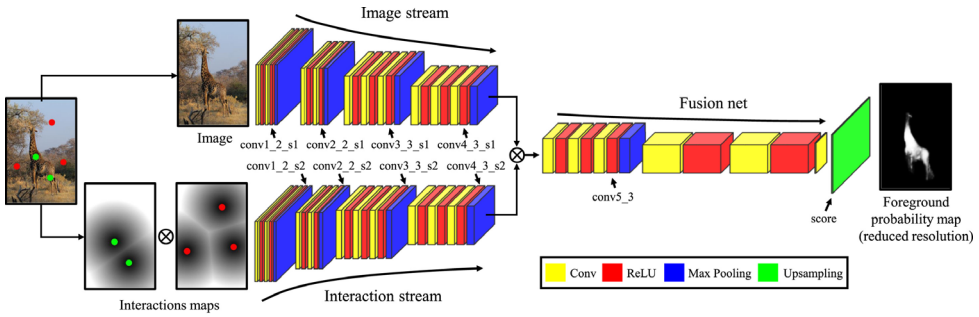
3.1 双流后期融合网络（TSLFN）

图 2（a）显示了所提出的 FCTSFN 中 TSLFN 的结构。 TSLFN 的输入有两部分：一部分是图像，另一部分是分别从正负用户交互产生的级联正负交互图（见 3.4 节）。 网络以降低的分辨率输出前景概率图，指示像素是前景的可能性。该网络使用 VGG16 网络（Simonyan＆Zisserman，2015）作为基础网络。它包括三个部分：图像流，交互流和融合网络。图像或交互流中的网络由 VGG16 网络的前 10 个卷积层组成，具有线性矫正单元（ReLU）。在几个卷积层之后，是内核大小为 2×2 且步幅为 2 的最大化层。两个流的目的是分别学习图像和交互图的深度特征。

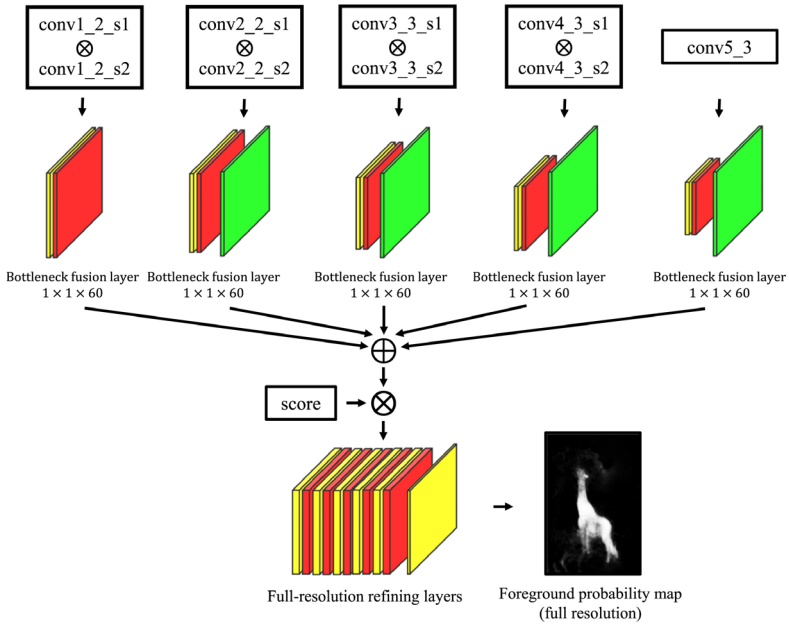
在图像和交互流的末尾，连接来自两个流的特征。然后应用融合网络学习组合连接的特征映射以预测前景/背景。 融合网络由 6 个卷积层组成：其中前 3 个来自 VGG16 网络的最后 3 个卷积层（对应于 VGG16 中的 conv5_1，conv5_2，conv5_3）; 最后 3 个使用 Shelhamer 等人（2017 年）的方法从 VGG16 网络中的完全连接层转移。由于整个网络的输出相对于输入图像被降低采样 32 倍，我们使用增采样层将输出升级到原始分辨率。 与 Shelhamer 等人类似。（2017），上采样层是反卷积层，滤波器设置为双线性插值核。

需要注意的是，可以设计出该 TSLFN 结构的变体。可以在图像/交互流和融合网络之间对 VGG16 网络的层进行不同的分配，以在两个流和融合网络中创建具有不同深度的 TSLFN 的变体。这实质上是用户交互的影响和网络的预测能力之间的权衡。如果我们在融合网络中使用较少的层，则用户交互中的位置信息可能对预测结果具有更高的影响，因为它减少了纯用户交互特征与预测结果之间的层数。然而，这可能损害网络的预测能力，因为融合网太浅并且可能无法学习从图像/用户交互特征到前景的有效映射。另一方面，更深的融合网可能具有足够的容量来学习从图像/用户交互特征到前景的映射，但是由于层之间的层数增加它可能削弱用户交互对预测结果的影响。在实验上，我们发现图 2（a）所示的结构与其他变体相比具有最高性能（见 4.2 节）。我们认为这是因为它在我们的基础网络中实现了用户交互影响和预测容量之间的最佳平衡。

此外，我们注意到融合图像和用户交互功能的另一种可能方式是 Hazirbas，Ma，Domokos 和 Cremers（2016）提出的逐层融合。该方法使用两个单独的流，并在不同层多次融合来自两个流的特征。我们已经在我们的实验的早期阶段用这种融合架构进行了实验（即，我们在我们的基础网络中执行图像和用户交互流之间的类似的逐层融合）。我们发现它导致了交互式图像分割任务的性能下降。考虑到该架构最初设计用于处理 RGB 深度（RGBD）数据，我们认为导致性能下降的数据特征的差异。对于 RGBD 数据，深度数据包括精确的对象边界信息，因此逐层融合图像数据增强了对象边界信息，如 Hazirbas 等人所述。 （2016）。然而，对于我们的交互式分割任务，交互图不包括这种准确的对象边界信息;有可能将交互流中的特征逐层融合到图像流中使得网络难以学习关于对象的信息。



(a) TSLFN



(b) MSRN

图 2. 提出的完全卷积双流融合网络（FCTSFN）的体系结构。 它包括一个双流后期融合网络（TSLFN）和一个多规模精炼网络（MSRN）。

3.2 多尺度精炼网络（MSRN）

所提出的 FCTSFN 中的 MSRN 旨在融合 TSLFN 中不同尺度的信息，以便更好地理解前景的位置并以全分辨率细化预测的前景。图 2（b）显示了 MSRN 的体系结构。 MSRN 使用来自 TSLFN 的六个不同比例的特征：在图像的每个合并层和交互流之前的级联特征映射（参见图 2 中的 conv1_2_s1，conv1_2_s2，conv2_2_s1，conv2_2_s2，conv3_3_s1，conv3_3_s2，conv4_3_s1，conv4_3_s2）；融合网中汇集层之前的特征映射（图 2 中的 conv5_3）；升级的预测分数（图 2 中的分数）。除了预测分数之外，每个比例的特征通过大小为 1*1*60 的卷积层（滤波器高度*滤波器宽度*滤波器数），并且具有下采样的特征映射被放大到原始分辨率（第二行）在图 2（b）中。我们将这些层称为“瓶颈融合层”，因为它们具有双重效应。首先，它们融合来自图像和交互流的特征图以在特定尺度上搜索前景的信息。其次，它们充当瓶颈层以减少特征图的维度以保持计算花费可以接受。

在瓶颈融合层之后，来自不同尺度的特征图通过逐元素求和操作融合。 然后，融合特征与来自 TSLFN 的预测得分连接。最后，连接的特征通过全分辨率细化层来预测精化的前景（图 2（b）中的最后一行）。全分辨率精炼层由六个卷积层的堆叠组成。这些卷积层的大小是 7*7*64, 5*5*64, 3*3*64, 3*3*64, 3*3*64, 1*1*2. 我们使用从大到小的过滤器来捕获从粗到细区域的

信息。最后一个卷积层用作分类器。

3.3 网络训练

我们分两个阶段训练 FCTSFN。在第一阶段，我们删除 MSRN 并从预先训练的 VGG16 基础网络中微调 TSLFN。 在第二阶段，我们修复 TSLFN 中的参数并从头开始训练 MSRN。

微调 TSLFN。我们在 ImageNet 数据集 Russakovsky 等，2015，Simonyan 和 Zisserman，2015 上预训练的 VGG16 网络上微调 TSLFN（图 2（a））。我们使用像素方式 softmax 损失。我们采用 Shelhamer 等人的“重”学习方案。（2017）使用批量大小和 0.99 的动量，由于报告了这种方法在微调 FCN 用于图像分割任务方面的有效性（Shelhamer 等，2017）。在这个阶段，由于形状的不同，预训练的权重不能直接应用于以下层：交互流中的第一个卷积层，融合网中的第一个卷积层，以及最后的卷积层。融合网。对于具有双通道输入的交互流中的第一卷积层，我们使用预训练的 VGG16 网络的第一卷积层中的滤波器的平均值来初始化它。由于来自两个流的特征图的串联，融合网络中的第一卷积层与 VGG16（conv5_1）中的对应层相比具有加倍数量的信道。为了初始化该层，我们将该层的通道分成两半，并将 vGG16 的 conv5_1 层中的预训练权重复制到每一半。对于融合网络中的最后一个卷积层，我们用全零来初始化它。此外，我们在 Shelhamer 等人中采用了类似的方法。（2017）为 TSLFN 微调步幅 16 网络和步幅 8 网络，以更精细的尺度来预测前景。我们使用 TSLFN 的 stride-8 网络作为 TSLFN 的最终形式，以使用 MSRN 进行预测。

从头开始训练 MSRN。通过训练和修复 TSLFN 参数，我们从头开始训练 MSRN。我们发现，在这个阶段，阶级失衡对训练效果有显着影响。具体来说，在交互式图像分割的任务中，前景通常占据比背景相对更小的区域。这导致背景像素远远多于训练数据中的前景像素（参见 4.1 节）。因此，学习的网络容易偏向背景，导致所有像素都是背景而没有前景的预测。为了解决这个问题，如果前景的边界框区域占据图像区域的 35％以下，我们会裁剪以前景为中心的图像。为避免过度拟合，我们在训练中使用了另外三种策略。 （1）数据增强：在前后传递之前，该传递中使用的训练图像有 50％概率旋转，以及接收随机翻译的 50％概率。 （2）丢弃：属于 MSRN 中的全分辨率精细化层的每个卷积层（参见图 2（b））后为丢失率为 0.5 的丢失层，除了用于分类的最后一层。 （3）早期停止：我们每 1000 次迭代记录验证数据的验证准确性，当我们观察到几次连续验证的准确性没有改善时，我们终止训练。 MSRN 训练的其他设置如下。我们使用 He，Zhang，Ren 和 Sun（2015）中的方法随机初始化所有卷积层。我们将所有训练图像的大小调整为 240 ＊ 320（高度*双倍）的分辨率，我们训练批量大小为 3.我们将初始学习率设置为 1e-8，权重衰减设置为 0.0005，动量设置为 0.99。

3.4 数据处理

本文中的数据处理包括两部分：用户交互图的生成和网络输出的后处理。 我们采用 Xu 等人的方法。（2016）从用户点击生成交互图作为网络的输入。 给定图像和用户点击，正和负点击的集合分别被转换为正和负交互图。交互地图与输入图像具有相同的高度和宽度。设$\mathcal{S}$是一组正或负点击。设$s_{ij} \in \mathcal{S}$是坐标(i,j)处的$\mathcal{S}$的点击。 设$Y_{m,n}$是对应于图像和点击的交互图中(m,n)处的元素。

$Y_{m,n}$计算方法是：

$$Y_{m,n} = \min_{s_{i,j} \in \mathcal{S}} \sqrt{(m-i)^2 + (n-j)^2}$$

换句话说，使用像素和用户点击之间的最小欧几里德距离来计算交互图。正负交互图中的像素值被截断为 255。如果没有接收到负点击，则负交互图中的所有像素值都设置为 255.正负交互图的示例包括在图 2（a）中。对于网络输出的后处理，我们采用类似于 Xu 等人的基于图切割的方法。（2016）。

4．实验

在本节中，我们将展示实验分析和所提方法的比较。 首先，我们描述实验设置。 然后，我们对提出的 TSLFN 和 MSRN 进行实验分析。 最后，我们与最先进的交互式图像分割算法进行了比较。

4.1 实验设定

数据集。 我们对四个数据集进行了实验：Pascal VOC 2012（Everingham，Gool，Williams，Winn，＆Zisserman，2010），Microsoft Coco（Lin et al。，2014），Grabcut（Rother et al。，2004）和 Berkeley（McGuinness＆ OConnor，2010）。 Pascal VOC 2012 和 Microsoft Coco 是用于对象分割的基准数据集。 Grabcut 和 Berkeley 是交互式图像分割的基准数据集。 对于 Pascal VOC 2012，我们使用其 1464 个图像和 1449 个图像的验证集的训练集； 对于 Microsoft Coco，我们从其 80 个类别中随机选择 20 个图像，类似于 Xu 等人的设置。（2016）； 使用拥有 50 张图像的 Grabcut 数据集和拥有 100 张图像的伯克利数据集中的所有图像。

数据分区。 我们将上述数据集中的数据划分为训练/验证/测试数据，如下所示。 我们使用 Pascal VOC 2012 训练集作为训练/验证数据。 从 1464 个图像中，我们随机选择 200 个图像作为验证数据，其余图像用作我们的训练数据。 我们使用训练数据来训练神经网络。 我们使用验证数据来监控和控制训练过程。由于收集真实用户的交互数据进行网络训练实际上太昂贵，我们采用 Xu 等人的方法。 （2016）为训练/验证数据中的对象生成合成用户交互。 我们使用除训练/验证数据之外的数据作为性能评估的测试数据（即 Pascal VOC 2012 验证集，Microsoft Coco，Grabcut，Berkeley）。

性能评估。我们使用前景的 IoU 来测量分割精度。它计算分割结果与真实掩模之间的交点与它们的并集的比率。基于 IoU，我们使用两种方法评估算法的性能：前景 IoU 与点击次数，以及实现特定前景 IoU 的点击次数。前一项衡量标准显示了用户点击次数的细分准确性;后一个方法显示了用户实现特定分割准确度的工作量。给定图像中的对象，我们会自动生成一系列点击作为用户交互（见下文）。我们跟踪前景 IoU 与点击次数，并记录实现特定前景 IoU 的点击次数。对于数据集，我们计算此数据集中所有对象的每个度量的平均值。在本文中，我们将 20 设置为最大点击次数。即如果在 20 次点击中无法实现某个 IoU，我们会将记录的点击次数阈值设置为 20。

生成点击序列。我们设计了一种方法来自动生成给定对象的一系列点击以进行性能评估。该方法在（1）基于当前分割结果向点击序列添加点击和（2）使用更新的点击序列更新分割结果之间进行迭代。给定当前分段掩模和背景实况掩码，我们按如下方式添加单击。首先，我们在当前的分割结果中找到错误正向和负向区域。然后，我们选择错误正向和负向区域中最大的连通成分。我们将点击放置在所选区域内且距离该区域边界最远的点上。如果将此单击放置在错误负向区域，则将其设置为正向单击，否则将其设置为负向单击。添加点击后，我们在评估中使用该算法来更新分段掩模。这种方法的意图是添加的点击集中在最大的误差区域，并尽可能地放置在该区域的中心部分。

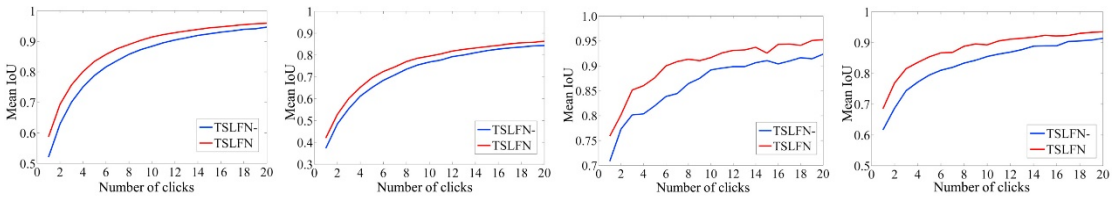
4.2 TSLFN 的评估

回想一下，我们在 TSLFN 中使用后期融合结构的意图是改善用户交互对预测结果的影响，因为用户交互是关于前景/背景位置的更准确信息。因此，作为本小节中的第一个实验，我们比较了用户点击对双流和单流网络之间预测结果的影响。衡量用户点击对预测结果的影响想法很简单：如果其周围区域在网络产生的前景概率图中具有更高的响应，则正点击具有更高的影响;如果负面点击的周围区域在前景概率图中具有较低的响应，则负面点击具有较高的影响。因此，如果正面和负面点击都会对网络输出产生很大影响，那么前景概率图中两种类型点击的响应应该具有良好分离的分布。因此，可以通过以下方式测量正和负点击的总体影响：（1）计算前景概率图中正和负点击周围区域中的响应的分布；（2）测量与正和负点击相对应的分布的分离程度。在本文中，我们采用可判定性指数（DI）来衡量两种分布之间的分离程度（Daugman，2004）：

$$DI = \frac{|\mu_p - \mu_n|}{\sqrt{\frac{\sigma_p + \sigma_n}{2}}}$$

其中μ_p和μ_n分别是正负点击响应分布的平均值;σ_p和σ_n表示两个分布的方差。

为了分析双流和单流网络之间用户点击的影响，我们比较了 TSLFN 与单流方法之间的 DI。为了将 TSLFN 修改为单流网络，我们删除了交互流，并将交互图与网络开头的图像连接起来。注意，此修改后的 TSLFN 等同于 Xu 等人的单流完全卷积网络（SSFCN）。（2016）。因此，我们将其称为 SSFCN。我们使用第 4.1 节中的方法自动生成 1,5 和 10 次用户点击来测量 DI（我们将此设置称为自由选择）。我们根据概率图中半径为 10 到正负点击的响应来计算 DI。如果不存在负面点击，我们会计算围绕正面点击的响应与所有背景区域中的响应之间的 DI。我们分别使用上述方法为每个对象计算 DI，并报告每个测试数据集的平均 DI。



(a) Pascal VOC 2012 (b) Microsoft Coco (c) Grabcut (d) Berkeley

图 3. 用于分析来自 TSLFN 的交互流的深度特征的影响的平均 IoU 与点击次数。横轴为点击次数，纵轴为 IOU

表 1. 概率图中正和负点击周围区域之间的可判定性指数（DI）（每个设置的粗体为最佳性能）。

数据集	自由选择		全正		单正	
	SSFCN	TSLFN	SSFCN	TSLFN	SSFCN	TSLFN
Pascal VOC 2012 (1 click)	12.97	**15.60**	12.97	**15.60**	-	-
Microsoft Coco (1 click)	16.79	**20.37**	16.79	**20.37**	-	-
Grabcut (1 click)	19.38	**21.38**	19.38	**21.38**	-	-
Berkeley (1 click)	71.42	**90.91**	71.42	**90.91**	-	-
Pascal VOC 2012 (5 clicks)	2.57	**6.40**	2.79	**4.03**	2.16	**2.92**
Microsoft Coco (5 clicks)	2.72	**9.61**	2.40	**3.58**	1.29	**2.10**
Grabcut (5 clicks)	3.84	**7.91**	3.14	**4.66**	1.89	**3.04**
Berkeley (5 clicks)	2.85	**5.22**	3.05	**4.33**	1.33	**2.23**
Pascal VOC 2012 (10 clicks)	1.24	**2.33**	2.33	**3.20**	2.34	**3.03**
Microsoft Coco (10 clicks)	1.20	**2.37**	2.06	**2.87**	1.30	**1.89**
Grabcut (10 clicks)	1.53	**3.16**	2.15	**3.09**	1.94	**3.06**

数据集	自由选择		全正		单正	
	SSFCN	TSLFN	SSFCN	TSLFN	SSFCN	TSLFN
Berkeley (10 clicks)	1.33	**2.46**	2.03	**2.88**	1.50	**2.21**

除了允许自由选择点击并导致积极和消极点击组合的自由选择设置外，我们还会在仅存在正面或负面点击时研究点击的影响。这是为了研究网络对不同类型用户点击的行为。为了研究只存在正面点击的情况，我们强制将所有点击放在前景（称为全正）;我们测量点击和整个背景区域周围区域之间的 DI。为了调查仅存在负面点击的情况，我们考虑一个近似设置：我们强制第一次点击在前景上，其余点击在背景上（称为单正）;我们测量负点击和整个前景区域周围区域之间的 DI。使用近似设置的原因是我们的所有训练数据都有至少一个正向点击给定生成它们的方法（参见第 4.1 节），因此网络没有经过良好训练，无需正面点击即可处理数据。

表 1 显示了四个数据集上的 DI，用户点击设置为自由选择，全正和单正。 我们可以看到，与 SSFCN 相比，TSLFN 具有更高的 DI。 这意味着在 TSLFN 的概率图中，正负点击响应的响应分布分离度更高。 换句话说，用户点击对双流网络结构具有更高的影响。 这与我们使用双流网络的目的一致。 此外，我们发现这种趋势适用于所有三种用户点击设置。 这表明双流网络实现的用户点击影响的改善对于不同类型的点击是一致的。

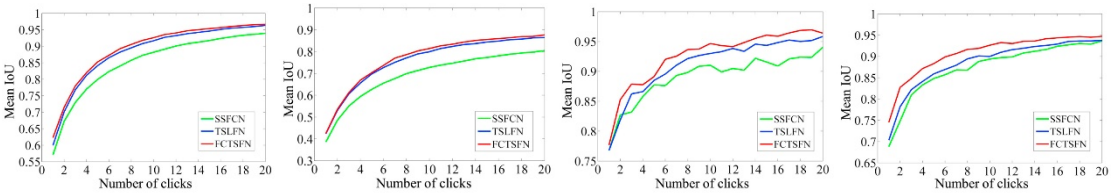
由于如上计算的 DI 基于正和负用户点击周围的区域，因此它们不代表整个前景区域和背景区域之间的响应的可分离性;因此，它们并不代表最终分割结果有多好。为了验证用户点击的更高影响是否有利于最终的细分性能，我们还需要比较 TSLFN 和 SSFCN 之间的细分准确性。请注意，在本文的其余部分，我们不限制用户点击的类型（即我们遵循上面的自由选择设置）。这有两个原因。首先，允许用户自由地发出正面和负面点击是最常见的情况。其次，自由选择设置实际上涵盖了所有积极和单积极的设置;例如，当前景非常小时，自由选择设置很可能产生与单正设置产生的点击序列相同的点击序列。如图 4，表 2，表 3 所示，与 SSFCN 相比，TSLFN 具有更好的最终分段性能。这一观察结果表明，通过双流网络架构改善用户交互对网络输出的影响，确实实现了改进的性能。

表 1 中另一个有趣的观察结果是，随着用户点击次数的增加，DI 通常会下降。我们认为有两个可能的原因。首先，用户只单击一次，网络就可以更专注于此次点击;这会导致点击周围的响应与背景中的响应之间存在非常大的差异，因此会导致较大的 DI。相反，随着用户点击次数的增加，网络可能会尝试在所有点击的影响之间取得平衡;这可能会导致每次点击周围的响应减少，从而降低 DI。其次，我们发现主要对象通常可以高精度地分割，只需很少的点击（见表 2）。因此，只需 5 或 10 次用户点击，对象边界附近可能会有很多点击。围绕这些点击的响应可能会降低整体 DI，因为前景概率图中的响应可能在对象边界周围较弱，因此它们在正和负点击之间的分离较少。

此外，仔细检查表 1 中的结果，我们可以发现，对于增加的用户点击次数，上述 DI 减少的观察仅适用于自由选择和全正设置。对于单正设置，DI 在 5 到 10 次点击之间非常相似。这一观察结果为网络行为带来了更有趣的可能性：网络竞争性地对待积极点击，同时平等对待负面点击。一方面，在全正设置的情况下，当点击次数增加时，DI 会减少。这可能意味着每个人的积极点击的影响之间存在竞争;它会导致每次正面点击的影响进行权衡，这会降低整体 DI 的积极点击次数。另一方面，对于单正设置，DI 在 5 到 10 次点击之间非常相似。这可能意味着添加负面点击几乎不会影响每个负面点击的影响。换句话说，网络平等对待每个负面点击。

上述实验验证了我们的想法，即双流网络允许用户点击对预测结果产生更大的影响，并且与单流网络相比，性能更高。但是，这又引出了另一个问题：我们是否需要用户交互的深层功能来应用这种影响？换句话说，TSLFN 所需的交互流是什么？为了证明为用户交互产生深层特征的交互流的

效果，我们比较了两个网络之间的性能：（1）TSLFN; （2）移除了交互流的 TSLFN（称为 TSLFN-）。具体地，在 TSLFN-中，交互图被调整大小并与图像流末尾的特征连接;然后将连接的特征用作融合网络的输入以预测前景(注意,这与 SSFCN 不同,其中交互图与整个网络开始处的原始输入图像连接)。图 3 比较了 TSLFN-和 TSLFN 之间的性能（注意，该图中的结果基于 stride-32 网络;最终的 stride-8 网络的性能可能类似，因为 stride-8 网络是基于 stride-32 网络)。可以看出，TSLFN-的性能下降。该结果表明，来自交互流的用户交互的深层特征对于 TSLFN 实现良好性能也是重要的。这可能是因为深度特征提供了更丰富和更有意义的用户点击标识，并且在与图像特征融合时可以更准确地指导分割过程。



(a) Pascal VOC 2012 (b) Microsoft Coco (c) Grabcut (d) Berkeley

图 4.用于分析 TSLFN 和 MSRN 的平均 IoU 与点击次数。横轴为点击次数，纵轴为 IOU

表 2. TSLFN 和 MSRN 为获得特定 IoU 的平均点击次数（以粗体表示的最佳性能）。

数据集	SSFCN	TSLFN	FCTSFN
Pascal VOC 2012 (85% IoU)	5.81	4.95	**4.58**
Microsoft Coco (85% IoU)	11.42	9.97	**9.62**
Grabcut (90% IoU)	5.02	4.28	**3.76**
Berkeley (90% IoU)	8.48	7.89	**6.49**

表 3. TSLFN 和 MSRN 网络，特定点击次数的平均 IoU（百分比，以粗体表示的最佳性能）。

数据集	SSFCN	TSLFN	FCTSFN
Pascal VOC 2012 (1 click)	57.0	60.0	**62.3**
Microsoft Coco (1 click)	38.6	42.3	**42.5**
Grabcut (1 click)	76.8	76.8	**77.7**
Berkeley (1 click)	68.8	70.3	**74.5**
Pascal VOC 2012 (3 clicks)	73.0	76.8	**78.0**
Microsoft Coco (3 clicks)	55.1	60.7	**61.2**
Grabcut (3 clicks)	83.2	86.3	**87.9**
Berkeley (3 clicks)	81.0	82.2	**84.8**
Pascal VOC 2012 (10 clicks)	88.2	91.7	**92.6**

数据集	SSFCN	TSLFN	FCTSFN
Microsoft Coco (10 clicks)	72.8	80.0	**81.5**
Grabcut (10 clicks)	91.0	93.0	**94.7**
Berkeley (10 clicks)	89.4	90.0	**92.6**

最后，我们介绍一些与我们提出的的 TSLFN 相关的实验结果。正如 3.1 节末尾所讨论的那样，在给定 VGG16 基础网络的情况下，我们可以在图像/交互流和融合网络中使用不同的深度来构建 TSLFN。具体而言，VGG16 基础网络具有 5 个 Conv-ReLU-Pool（CRP）块。我们在图 2（a）中的 TSLFN 结构使用前 4 个 CRP 块来形成图像/交互流，并且它使用 VGG16 的其余部分作为融合网络。可以通过使用不同数量的 CRB 块来形成图像/交互流来创建该架构的变体。这导致所提出的 TSLFN 的变化在图像/交互流和融合网络中具有不同的深度。我们使用 TSLFN_来表示所提出的 TSLFN 与用作图像/交互流的基础网络中的第一个 CRP 块的变化。图 2（a）中所示的所提出的 TSLFN 基本上等同于 TSLFN_4。它有四种变体：TSLFN_1，TSLFN_2，TSLFN_3，TSLFN_5。在这些变化中，TSLFN_1 具有最浅的图像/交互流和最深的融合网，而 TSLFN_5 具有最深的图像/交互流和最浅的融合网。

图 5，表 4，表 5 显示了所提出的 TSLFN 的所有上述变化的性能。 可以看出，所提出的 TSLFN（图 5 中的 TSLFN_4，表 4，表 5）通常在其变体中具有最高性能。 可能的原因是我们在第 3.1 节末尾讨论的那个：在用户交互的影响和图像/交互流和融合网络中不同深度的预测容量之间存在权衡；在给定我们的基础网络的情况下，如图 2（a）所示的所提出的 TSLFN 结构与其它因素相比实现了两个因素之间的最佳平衡。

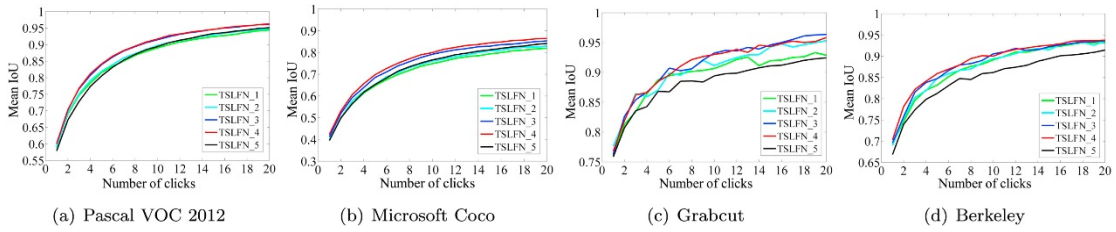


图 5. TSLFN 的平均 IoU 与点击次数。横轴点击次数，纵轴 IOU

表 4. TSLFN 实现特定 IoU 的平均点击次数（以粗体表示的最佳性能）。

数据集	TSLFN_1	TSLFN_2	TSLFN_3	TSLFN_4	TSLFN_5
Pascal VOC 2012 (85% IoU)	5.63	5.43	**4.95**	**4.95**	5.73
Microsoft Coco (85% IoU)	11.06	10.83	10.18	**9.97**	10.99
Grabcut (90% IoU)	4.60	4.66	4.44	**4.28**	5.14
Berkeley (90% IoU)	8.21	8.57	8.15	**7.89**	9.27

表 5. TSLFN 特定点击次数的平均 IoU（百分比，粗体表示最佳性能）。

数据集	TSLFN_1	TSLFN_2	TSLFN_3	TSLFN_4	TSLFN_5
Pascal VOC 2012 (1 click)	59.3	58.9	**60.1**	60.0	57.9
Microsoft Coco (1 click)	39.7	40.0	41.2	**42.3**	39.5
Grabcut (1 click)	76.1	**77.6**	76.3	76.8	75.9
Berkeley (1 click)	69.3	69.0	69.6	**70.3**	66.8
Pascal VOC 2012 (3 clicks)	74.7	75.1	76.7	**76.8**	72.8
Microsoft Coco (3 clicks)	56.4	57.7	59.3	**60.7**	56.5
Grabcut (3 clicks)	83.4	**86.4**	85.4	86.3	83.6
Berkeley (3 clicks)	79.6	80.3	81.6	**82.2**	77.4
Pascal VOC 2012 (10 clicks)	89.0	89.8	91.5	**91.7**	89.4
Microsoft Coco (10 clicks)	74.9	76.1	79.2	**80.0**	76.6
Grabcut (10 clicks)	90.7	91.2	**93.2**	93.0	89.4
Berkeley (10 clicks)	89.1	89.4	**90.6**	90.0	86.2

在本小节中，我们评估了 TSLFN。 结果证实：（1）与单流网络相比，TSLFN 的双流结构允许用户点击的信息对网络输出产生更大的影响，从而带来更好的性能；（2）从用户交互中提取深层特征对于 TSLFN 实现更好的性能也很重要。 我们还验证了所提出的 TSLFN 的设计选择，表明它通常在给定基础网络的所有变化中性能最佳。

4.3 MSRN 评估

为了分析 MSRN 的影响，我们比较了两个网络之间的性能：TSLFN 和 FCTSFN（即 TSLFN + MSRN）。通过比较图 4，表 2，表 3 中报告的性能，我们可以看出 FCTSFN 具有始终比 TSLFN 更好的性能。 这些观察验证了 MSRN 利用多尺度特征来细化分割结果的有效性。 我们认为，有两个可能的原因改善了性能。 第一，MSRN 以全分辨率进行预测，因此在对象边界处更准确。 其次，MSRN 从网络的开始到结束都在使用特征。 因此，它将来自颜色/边界等低级特征的信息融合到具有对象级理解的高级特征中；这使网络能够对前景和背景建立更全面的理解，并且可以获得更准确的分割结果。

4.4 与目前算法比较

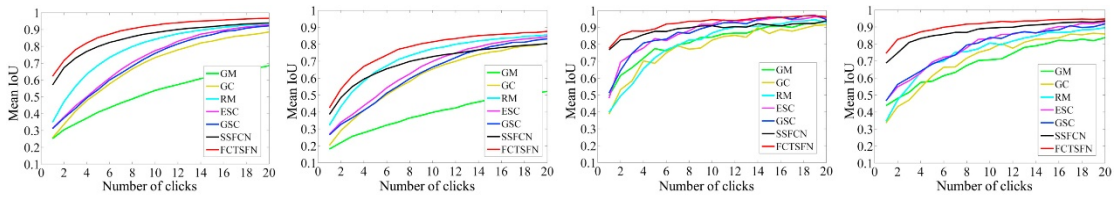
在本小节中，我们将建议的网络与最先进的算法进行比较。我们将比较分为两部分：限制比较和不受限制的比较。在限制性比较中，我们严格按照 4.1 节中的实验设置进行实验； 我们运行对照方法的代码，或自己实现。在无限制的比较中，我们直接与已发表论文中性能指标进行比较。 需要注意，在无限制比较中，由于不同论文中实验设置的差异，结果不完全可比。 然而，它显示了所提出的网络在最先进的方法中的性能，并且对实验设置进行了开放选择。

受限制的比较。对于限制性比较，我们将比较以下方法：图形切割（GC）（Boykov＆Jolly，2001），测地线消光（GM）（Bai＆Sapiro，2007），随机游走（RW）（Grady，2006），欧几里德 星形凸（ESC）（Gulshan 等，2010），测地星凸（GSC）（Gulshan 等，2010），和单流 FCN（SSFCN）（Xu 等，2016）。

图 6 显示了测试数据上所有四个数据集的所有比较方法的平均 IoU 与点击次数。表 6 显示了某些特定点击次数（1,3,10）的平均 IoU。可以看出，与其他方法相比，所提出的 FCTSFN 实现了改进的性能。具体来说，在 Pascal VOC 2012，Microsoft Coco 和 Berkeley 数据集上，FCTSFN 与其他方

法相比表现更好。在 Grabcut 数据集上，当点击次数低于 10 时，FCTSFN 可以获得更好的性能;当点击次数大于 10 时，FCTSFN 的表现与 ESC 和 GSC 相似，并且与其他方法相比具有更好的性能。FCTSFN 在 Pascal VOC 2012，Microsoft Coco 和 Berkeley 数据集上显示出比在 Grabcut 数据集上更大的优势。一个可能的原因是 Grabcut 数据集具有较少数量的图像，具有更明显的前景/背景。因此，给定足够的点击次数，FCTSFN 在 Grabcut 数据集上与 ESC 和 GSC 的表现类似。总之， FCTSFN 在更大和更具挑战性的数据集上显示出始终如一的改进性能，同时它仍然在较小且不太具有挑战性的数据集上实现稳定和最佳性能。表 7 报告了实现某个 IoU 的平均点击次数。可以看出，提议的 FCTSFN 需要对所有数据集进行最少的点击次数。实现特定 IoU 的最佳点击次数为 1。因此，提议的 FCTSFN 在 VOC 2012 数据集上实现了相对于 SSFCN 的最佳性能改进为$(5.81-4.58)/(5.81-1)\approx 25.6\%$。 对于 Microsoft Coco，Grabcut 和 Berkeley 数据集，此数字分别为 23.3%，31.3%和 26.6%。 图 7 显示了给定相同用户点击的不同方法的一些示例结果。 图 8 显示了所提出的方法在测试数据集中的不同对象上的一些示例结果，其中自动生成的点击序列最多具有 5 次点击。

无限制的比较。对于不受限制的比较,我们将与以下方法进行比较:RIS-Net(Liew 等人,2017)，DEXTR（Maninis 等人，2018）和潜在多样性网络（LDN）(Li 等人，2018)。 我们直接引用点击次数来实现这些论文中报告的某个 IoU。 注意，如上所述，由于实验设置的不同，这些结果不能直接比较。 例如，对于 Microsoft Coco 数据集上的测试数据，不同的方法使用不同的随机采样设置; 这些方法还采用不同的训练数据和各种训练策略; 然而，这种比较显示了所提出的网络在具有开放式实验设置选择的最先进方法中的性能。



(a) Pascal VOC 2012 (b) Microsoft Coco (c) Grabcut (d) Berkeley

图 6. 平均 IoU 对比限制比较的点击次数。横轴点击次数，纵轴 IOU

表 6. 限制性比较的特定点击次数平均 IoU（百分比，粗体代表最佳表现）。

数据集	GC	GM	RW	ESC	GSC	SSFCN	FCTSFN
Pascal VOC 2012 (1 click)	25.2	25.7	34.9	31.4	31.2	57.0	**62.3**
Microsoft Coco (1 click)	18.1	20.3	32.3	26.9	26.6	38.6	**42.5**
Grabcut (1 click)	49.6	38.6	39.8	48.2	51.2	76.8	**77.7**
Berkeley (1 click)	43.8	33.6	34.6	46.7	46.7	68.8	**74.5**
Pascal VOC 2012 (3 clicks)	33.9	41.4	56.1	44.6	43.1	73.0	**78.0**
Microsoft Coco (3 clicks)	25.7	35.2	52.0	38.5	36.7	55.1	**61.2**
Grabcut (3 clicks)	66.2	58.4	56.1	74.4	74.6	83.2	**87.9**
Berkeley (3 clicks)	51.6	47.1	55.0	58.4	60.1	81.0	**84.8**

数据集	GC	GM	RW	ESC	GSC	SSFCN	FCTSFN
Pascal VOC 2012 (10 clicks)	53.8	73.3	84.2	77.5	75.9	88.2	**92.6**
Microsoft Coco (10 clicks)	39.8	65.4	77.4	69.9	66.6	72.8	**81.5**
Grabcut (10 clicks)	84.3	82.5	86.8	91.8	91.0	91.0	**94.7**
Berkeley (10 clicks)	70.8	76.9	80.6	83.0	83.8	89.4	**92.6**

表 7.针对限制性比较实现特定 IoU 的平均点击次数（粗体代表最佳性能）。

数据集	GC	GM	RW	ESC	GSC	SSFCN	FCTSFN
Pascal VOC 2012 (85% IoU)	14.81	10.59	7.98	8.22	8.48	5.81	**4.58**
Microsoft Coco (85% IoU)	17.74	14.57	11.71	11.70	12.11	11.42	**9.62**
Grabcut (90% IoU)	9.70	9.26	10.28	5.84	5.02	5.02	**3.76**
Berkeley (90% IoU)	13.68	14.10	13.46	9.73	9.38	8.48	**6.49**

表 8 表现了无限制比较中所有方法的性能。可以看出， FCTSFN 在 Pascal VOC 2012，Grabcut 和 Berkeley 数据集上实现了竞争性能。具体而言，与 RIS-Net 相比，FCTSFN 需要更少的点击在 Pascal VOC 2012 和 Grabcut 数据集上实现某个 IoU;它需要比 RIS-Net 多 0.46 次点击才能在 Berkeley 数据集上实现 90% IoU。与 DEXTR 相比，FCTSFN 在 Grabcut 数据集上表现更好，并且需要 0.58 次点击才能在 Pascal VOC 2012 数据集上实现 85% IoU。与 LDN 相比，FCTSFN 在 Grabcut 数据集上实现了更好的性能。请注意，与提议的 FCTSFN 相比，DEXTR 通过更多的训练数据实现了报告的性能（Pascal VOC 2012 + SBD Hariharan，Arbelaez，Bourdev，Maji，&Malik，2011），以及在线硬件示例挖掘（OHEM）（Shrivastava， Gupta，&Girshick，2016）基于训练策略和更先进的基础网络（ResNet-101 He et al.，2016）;类似地，LDN 通过更大的训练集（SBD）和更先进的分割网络（上下文聚合网络 Chen，Xu 等人，2017，Yu 和 Koltun，2016）实现其报告的性能。相比之下，FCTSFN 在表 8 中以较少的训练数据（仅 Pascal VOC 2012）实现了性能，同时在训练过程中没有对训练数据进行硬挖掘，并且使用不太先进的基础网络（VGG16）。

表 8. 无限制比较实现特定 IoU 的平均点击次数（粗体表示最佳性能）。

数据集	RIS-Net	DEXTR	LDN	FCTSFN
Pascal VOC 2012 (85% IoU)	5.12	**4.00**	-	4.58
Microsoft Coco (85% IoU)	-	-	**7.89**	9.62
Microsoft Coco seen categories (85% IoU)	5.98	-	-	-
Microsoft Coco unseen categories (85% IoU)	6.44	-	-	-
Grabcut (90% IoU)	5.00	4.00	4.79	**3.76**
Berkeley (90% IoU)	**6.03**	-	-	6.49

另一方面，从表 8 可以看出，与 RIS-Net 和 LDN 相比， FCTSFN 需要最多的点击次数来实现 Microsoft Coco 数据集上的 IoU。 但是，由于每种比较算法采用不同的随机采样设置，我们无法估计这种采样对最终性能的影响。 例如，我们在 Microsoft Coco 测试数据上实现的 SSFCN（如表 7 所示）性能低于其他方法中对于 Microsoft Coco 数据集的测试数据 Li et al.，2018，Liew 等人报告的性能。2017 年。

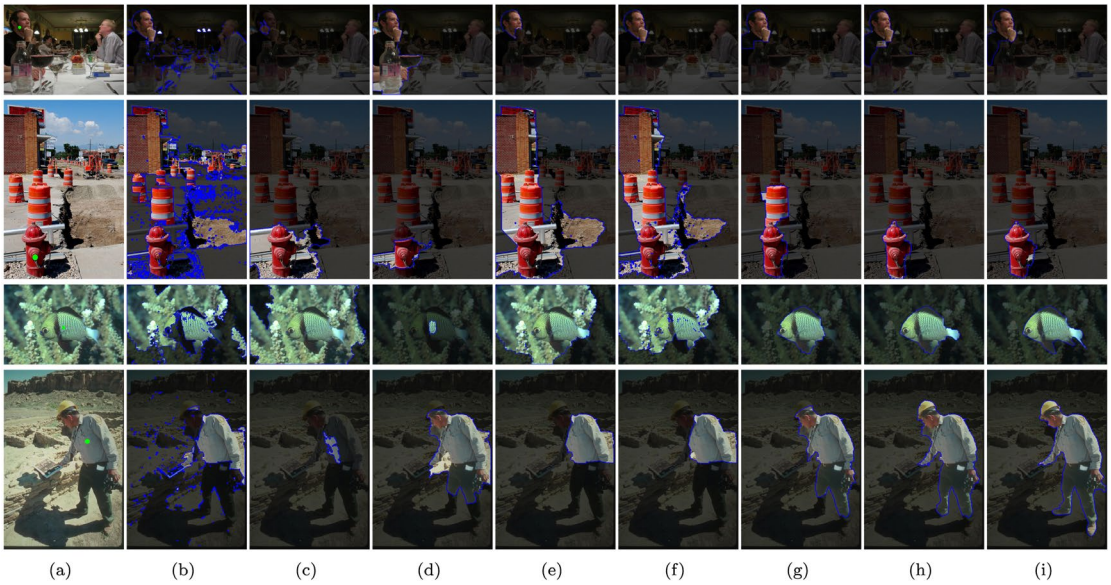


图 7.给出相同用户交互不同方法的分割结果的示例。 （a）用户点击的原始图像；（b）GC；（c）GM；（d）RW；（e）ESC；（f）GSC；（g）SSFCN；（h）FCTSFN；（i）基本事实。

在本小节中，我们介绍了，与限制性比较中的其他比较方法相比， FCTSFN 实现了改进的性能。在不受限制的比较中，与其他方法相比，它还可以通过较少的训练数据和不先进的网络实现能够匹敌的性能。

图 8. 所提出的方法对具有自动生成的点击序列的测试数据中的不同对象的分割结果的示例；每行从左到右：点击次数从 1 增加到 5。

5. 结论

在本文中，我们提出了一种新的完全卷积双流融合网络（FCTSFN）用于交互式图像分割。 目的是首先使用双流后期融合网络（TSLFN），以允许用户交互对分割结果具有更直接和更高的影响，以实现更高的准确性，然后使用多规模精炼网络（MSRN）来细化得到全分辨率的分割结果，以解决 TSLFN 中的分辨率损失。我们对四个基准数据集进行了全面的实验分析和比较。主要研究结果总结如下：

- 我们通过实验验证 TSLFN 中的双流结构允许用户交互对分段结果产生更大的影响，并且与单流网络相比，它实现了改进的性能。

- 我们通过实验验证了 TSLFN 中交互流的重要性：具有交互流的 TSLFN 比没有该流的 TSLFN 表现更好。 这意味着所提出的网络中的交互流成功地从各个用户交互数据中学习得到更丰富且更有意义的特征。

- 我们通过实验验证了所提出的 TSLFN 的设计。在给定固定基础网络的情况下，与其变体相比，所提出的体系结构通常具有更好的性能。

- 我们通过实验验证 FCTSFN 中的 MSRN 的前景精炼使 TSLFN 性能的进一步改善。

- 在有限的比较中，与最先进方法相比， FCTSFN 实现了更好的性能。

- 在不受限制的比较中，与最先进方法相比， FCTSFN 还能以较少的训练数据和不太先进的基础网络实现相匹敌的性能。

未来的工作可能侧重于：（1）利用更先进的基础网络实现双流结构，以实现更好的性能; （2）进行更多的实验和理论分析，以深入了解交互式图像分割的双流网络结构。

致谢

这项工作得到了 Ice Communication Limited 和 Innovate UK（项目 KTP / 10412）的支持。