

**显著性物体检测**

院 系 信息工程学院

专　 　 业 电子信息工程

年 级 2020级

授 课 教 师 刘铁

论 文 作 者  第二组成员

完 成 日 期 2023年6月28日

**目录**

[一、 **研究背景介绍** 3](#_Toc137670354)

[二、 **研究问题定义** 4](#_Toc137670355)

[三、 **研究数据介绍** 4](#_Toc137670356)

[四、 **研究算法介绍** 6](#_Toc137670357)

[1. 算法解决的问题 6](#_Toc137670358)

[2. 核心算法介绍 6](#_Toc137670359)

[3. 注意力引导跟踪模块 7](#_Toc137670360)

[4. 自适应像素强度损失函数 9](#_Toc137670361)

[五、 **算法结果展示与分析** 11](#_Toc137670362)

[1. 定性分析 11](#_Toc137670363)

[2. 基于六种评价指标分析 13](#_Toc137670364)

[六、 **小组成员贡献** 14](#_Toc137670365)

# 研究背景介绍

显著性物体检测最早可以追溯到1998年，Itti等发表了第一个基于显著度的视觉主义计算模型，使用多种特征的融合获得最终的显著图，视觉注意力机制模型正式引入到计算机视觉领域，从此显著性物体检测算法得到了蓬勃发展。现在显著性物体检测可以分为两种传统的显著性物体检测方法和基于深度学习的显著性物体检测方法。其中传统显著性物体检测算法还可以继续划分为基于对比度的显著性检测算法、基于频域的计算方法、基于稀疏理论的计算方法、基于图模型的计算方法、基于背景和前景先验的计算方法。在这其中例如Cheng等人提出全局对比度算法HC和RC等。在传统的方法使用空域或领域中大量的特征对比或者利用先验信息等进行图像显著性检测，其针对性较强，一般来说只适用于主题明确、物体颜色及结构较为单调的场景。基于深度学习的显著性物体检测方法又可以分为基于传统卷积神经网络的方法和基于完全卷积神经网络的方法。基于传统卷积神经网络的方法是将图像块的区域大小调整到固定的尺寸或进行多尺度的处理，然后进行卷积神经网络的特征提取，最后融合得到显著图。基于完全卷积神经网络的方法主要是利用VGGNet骨干网络架构和ResNet骨干网络架构。在基于深度学习的显著性物体检测方法中主要是利用卷积、下采样和上采样，会丧失显著物体的一些细节，在此基础上基于复杂的特征融合网络结构被提出。以下是显著性物体检测的发展历程图：

日程表

描述已自动生成

图 1 发展历史

为了提高显著目标检测（SOD）的性能，现有方法可以分为两种，分别是改进边缘表示的方法和减少多级聚合过程中差异的方法。但是由于在显著物体检测的过程中，存在目标背景复杂，边缘细节不够突出，算法复杂程度较高，计算效率低等等的问题，目前所提出的显著物体检测始终还不能很好地接近我们人工判断的理想的显著物体检测目标理想图。文献[1]提出了一个简单而强大的深度网络架构，U2NET，用于显著目标检测（SOD），其架构是一个两级嵌套的U结构。由于其中混合了不同大小的感受野，所以能够从不同尺度捕获更多的上下文信息；并且由于其中的U结构中使用了池化操作，增加了整个架构的深度，而不会显著增加计算成本，这种U2NET结构能够从头开始训练深度网络，而无需使用图像分类任务的主干。文献[2]提出了一种自适应和专注深度蒸馏器，A2dele，用于高效的RGB-D突出物体检测。由于现有的最先进的RGB-D突出目标检测方法依赖于双流架构探索RGB-D数据，其中需要独立的子网来处理深度数据，不能避免会产生额外的计算成本和内存消耗，并且在测试过程中使用深度数据可能会阻碍RGB-D显著性检测的实际应用，通过使用A2dele来探索使用网络预测和注意力作为将深度认知从深度流转移到RGB流的桥梁的方式来提高显著物体检测的性能。但是这种方法在分流上所耗费的时间成本比较高，大大提升了训练的成本，并且所得到的结果与现有的先进方法对比所达到的性能效果并不高。

# 研究问题定义

显著性物体检测可以分为静态物体检测和动态物体检测，静态物体检测主要是针对一幅图像中的显著物体进行检测，而动态显著物体检测主要是对视频场景中的显著物体进行检测。一般来说，对于视频中的运动物体，其显著性检测依赖于每一帧图像，当存在物体消失和出现时，其显著性检测也依赖于前后帧图像的对比。因此，图像的显著性物体检测是基础，所以本次大作业对显著性物体检测方法是在静态图像的显著性物体检测上。对于静态图像的显著性物体检测的研究问题是根据任意给定一张图片，检测出图片中的明显物体，然后就是检测出的显著图应具有高分辨率，同时能够准确定位并保留原始图像的信息，最后就是如何快速检测出显著物体。

# 研究数据介绍

我们在整个实验的过程中，我们训练所用的数据集是DUTS-TR、测试的数据集有三个分别是：ECSSD、DUTS-TE、DUT-OMRON。以下是对数据集的基本介绍说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据集名称 | 图片数量 | 物体属性 | 物体特点 | 发表年份 |
| DUT-OMRON | 5168 | 多目标，大小不一 | 复杂 | 2013 |
| ECSSD | 1000 | 多为单目标，目标较大 | 简单 | 2015 |
| DUTS | 10，533 | SOD最大的基准数据集 | 复杂 | 2017 |

表 1 数据集说明

在这些数据集中DUT-OMRON， ECSSD，DUTS被广泛应用于深度显著性检测模型的训练，下图展示了这3个常用数据集的图片和对应的真值。

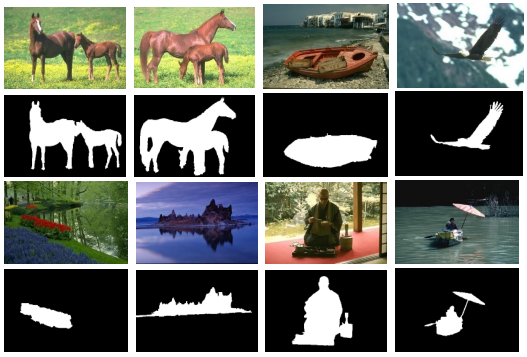
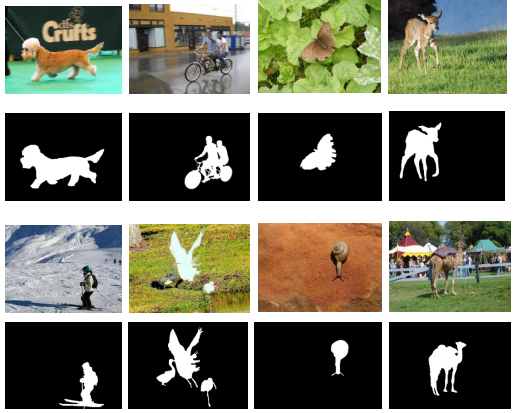
 



图 2 3个常用数据集图片与真值（左上为ECSSD数据集、右上为DUTS-TE数据集、下为DUT-OMRON数据集）

在本次大作业中我们测试所用的的数据集是DUTS-TE，该数据集来自DUTS是Wang等于2017年建立的，该数据集包含10553张训练数据和5019张测试数据，总共15572张图片，适用于基于深度学习的检测方法，需要大量标签的数据。该数据集的图片中具有多个目标，并且目标与背景相似度高，被广泛应用于基于深度学习的方法当中。

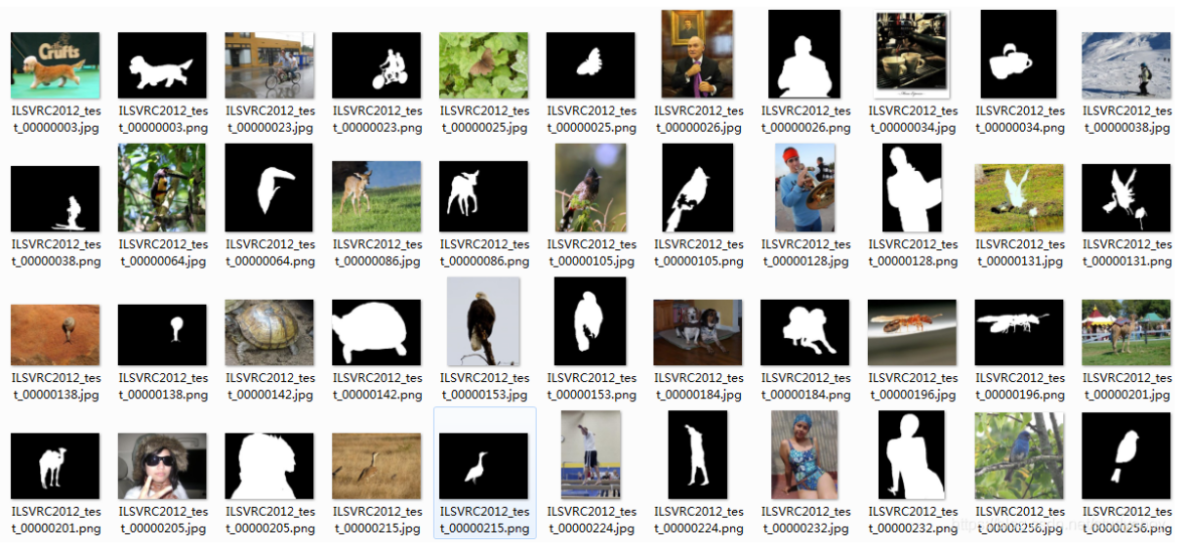


图 3 DUTS图片与真值

# 研究算法介绍

本次大作业的算法模块的实现是基于《TRACER: Extreme Attention Guided Salient Object Tracing Network》这篇论文来完成的，它通过结合注意力引导跟踪模块来检测具有明确边缘的突出物体。我们在第一个编码器的末端使用了屏蔽边缘注意力模块，适用快速傅里叶变换将细化的边缘信息传播到下游特征提取。在多级聚合阶段，联合注意力模块识别互补通道和重要空间信息。为了提高解码器的性能和计算效率，我们使用对象注意力模块最大限度减少解码快的适用。该模块从细化的通道和空间表示中提取未检测到的对象和边缘信息。随后，我们提出一种自适应像素强度损失函数来处理相对重要的像素，这与平等对待所有像素的传统损失函数不同。与3种现有的方法的比较表明，TRACER在三个基准数据集上实现了较先进的性能。

## 算法解决的问题

现有的显著性目标检测模型研究由于明确的边缘信息即兴发挥了SOD性能，但它们广泛使用边缘细化模块，这些模块需要全部或部分编码器输出。这些方法增加了开销和计算效率低下。此外，这些研究中的方法没有在下游特征提取阶段利用精细的边缘，因为边缘生成方法依赖于其它编码器输出。由于低级和高级信息对于确定边缘特征和语义表示是有效的，因此应有选择地确定这两个级别的细化模块的使用，并尽量减少网络效率。

## 核心算法介绍

本次大作业中算法提出了TRACER，为了解决现有方法的低效率问题，我们在浅层编码器、多级聚合过程和解码器中应用了三个注意力引导模块（即屏蔽边缘、联合和对象衰减模块）。屏蔽边缘衰减模块使用快速 傅里叶变换增强低级表征中的边缘特征，并将边缘优化表示传播到下一个编码器。联合注意力模块聚合多级编码器输出，以减少分布中的差异。随后，该模块确定聚合门控通道和空间表示中和解码器输出，以识别突出的物体。为了处理像素的相对重要性，我们提出了一个自适应像素强度损失函数。通过聚合多核聚合来聚合目标像素周围的相邻像素，并排除边缘外的权重。当目标像素由精细或明确的边缘组成时，为其分配的强度高于其他像素。以下是该算法的体系结构：

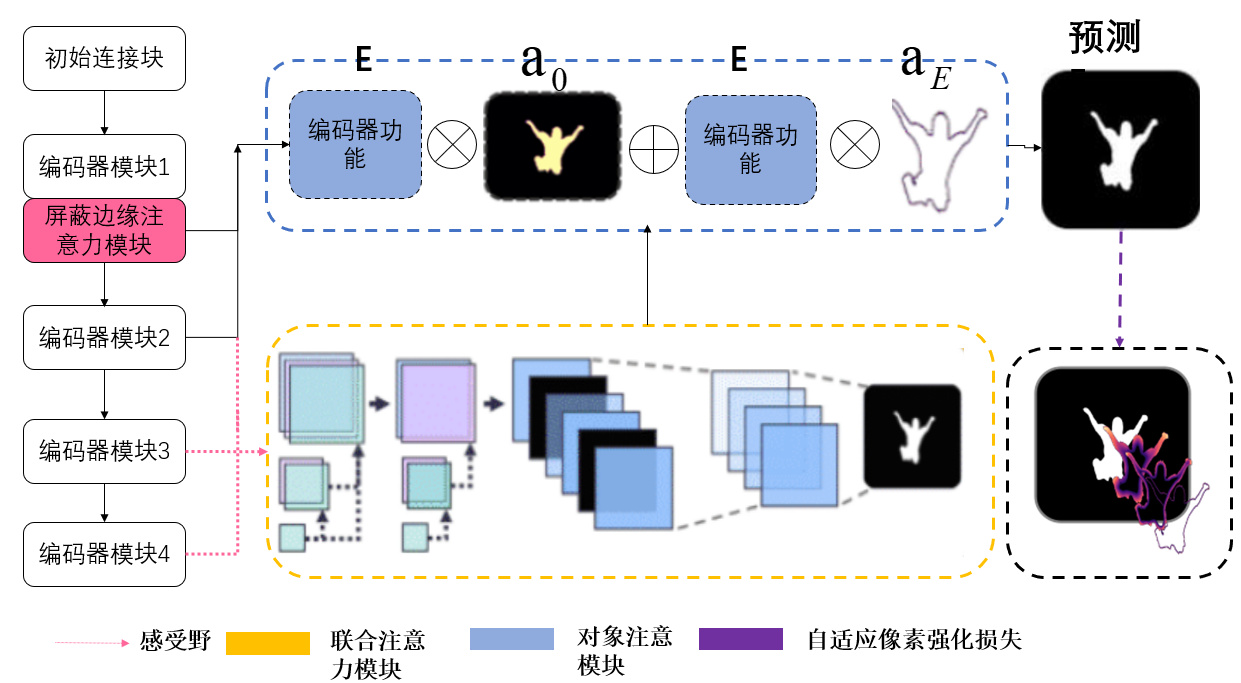


图 4 算法完整结构

针对该算法，我们首先来简单介绍一下该算法的实现过程。它通过结合注意力引导跟踪模块来检测具有明确边缘的突出物体，在第一个编码器的末端使用了屏蔽边缘注意力模块，适用快速傅里叶变换将细化的边缘信息传播到下游特征提取。在多级聚合阶段，联合注意力模块识别互补通道和重要空间信息。为了提高解码器的性能和计算效率，我们使用对象注意力模块最大限度减少解码快的适用。该模块从细化的通道和空间表示中提取未检测到的对象和边缘信息。随后，我们提出一种自适应像素强度损失函数来处理相对重要的像素，从而来提高性能。

## 注意力引导跟踪模块

检测具有边缘的不同物体对于提高SOD性能至关重要。使用卷积模块（为提高计算效率），通过注意力引导的突出对象跟踪模块（ATM）跟踪对象和边缘以提高性能。

**3.1 屏蔽边缘注意力模块**

为了跟踪边缘信息，本文提出了屏蔽边缘注意力模块（MEAM），通过使用快速傅里叶变换（FFT）提取显式边界，建立第一个编码器的输出边界。现有的方法使用边缘信息，但是它们无法在特征提取阶段利用显式边缘，由于这些方法需要深度编码器的输出才能获得不同的边缘。因此，我们使用FFT仅从第一个编码器表示中提取显式边缘。同时使用傅里叶变换及其逆变换，将第一个编码器表示为高频率和低频率，如下所示：

 （1）

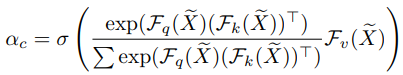
其中X表示输入特征，是一个高通滤波器，它消除了除半径r以外的所有频率，为了判别显式边缘，我们利用了高通滤波器获得的高频，这些滤波器具有足够的边界信息。此外，当包含了背景噪声，我们采用接受场运算RFB来消除噪声。通过感受野操作生成显式边缘，最后计算精细边缘，明确边缘损失。

**3.2 联合注意力模块**

联合注意力模块的（UAM）的目的是聚合多层次的功能，并检测更重要的上下文通过和空间表示。在这里，和分别表示卷积运算和通道特性链接。每个编码器输出分别聚合到32、64和128个通道，积分如下：

 （2）

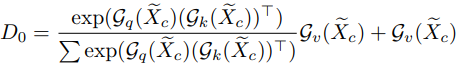
得到的一个聚集表示，通过聚合后，仍然是上下文信息是相对显著的信道和空间特征。然而，现有的研究已经将通道和空间注意力模块独立于解码器和感受野块，尽管这两个空间的依赖性。因此，我们同时要强调空间信息的基础上获得互补的信息通道的上下文。

 （3）

Xe∈Rc×1×1是信道分组表示，F(·)表示使用1×1核大小的卷积运算。有效通道αc∈Rc×1×1，得到上下文信息。为了细化聚合表示X，我们应用置信通道权重如下：xc=(X⊗αc)+X。随后，根据αc的分布和置信比γ保留信任通道，如下所示：

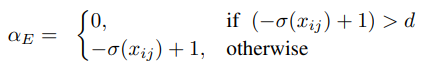
 （4）

这里，表示αc的γ分位数。我们排除了分布αc下部尾部的γ区域。然后，在空间上计算精炼的输入Xec以区分显著对象并产生第一解码器表示，，如等式5所示。

 （5）

**3.3 对象注意模块**

为了减少使用最小参数的编码器和解码器表示之间的分布差异，我们组织了一个对象注意模块（OAM）作为解码器。与现有的研究（Chen et al.2018；Zhao et al.2019）相比，我们将D作为解码器效率的单一通道，并且OAM跟踪来自每个解码器表示的对象边缘和互补边缘。同时进行细化突出物操作，由于不能检测到整个对象与明确的边缘区域，因此，我们生成一个互补的边缘权重去覆盖未检测到的区域。对于D中的每个像素我们反转检测到的区域，并消除与缺失区域检测的去噪比相对应的背景噪声：

 （6）

我们结合编码器输出和解码器特征：

 （7）

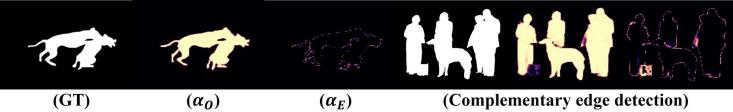
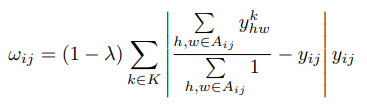
为了减少差异，我们利用感受野操作RFB（·）和上采样Di+1产生DSi+1。

图 5：物体互补边缘检测

## 自适应像素强度损失函数

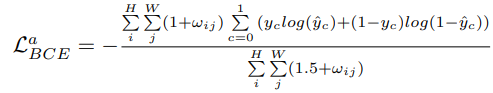
对于损失函数，我们结合二进制交叉熵（BCE），LoU，和L1损失函数来减少对象和背景之间的差异。虽然二进制交叉熵和LoU在全球范围内采用的损失函数，这些功能会导致类之间的差异的前景和背景时，所有的像素被认为是平等的。与显著对象的背景和中心的像素相比，与精细或显式边缘相邻的像素需要更多的关注。因此，我们提出自适应像素强度（API）损失，将像素强度应用于每个像素，如下所示：

 （8）

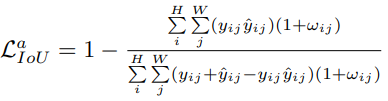
我们通过使用多个内核大小K并排除边缘之外的权重，聚合目标像素周围的相邻像素。如果目标像素由细边缘组成，则采用多核聚合来为目标像素分配比其他像素更多的权重。如下图所示：

图 6：对应内核大小K的像素强度w可视化

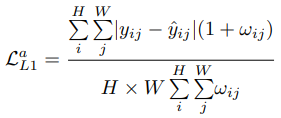
像素强度用于自适应损失函数损耗。通过使用像素强度和自适应损失函数使得网络更多地关注与显式或细边相关的局部结构，而不像BCE损失。

 （9）

与此相反，自适应的LoU损失优化的全局结构的基础上的密集特征对应ω。如（10）所示，与原始的loU损失相比，对于密集区域高度关联的像素进行了区分和强调。

 （10）

此外，为了进一步提高网络的等方差学习，以减少分歧的差异，我们测量的L1距离，这使网络学习强劲对嘈杂的标签（Ghosh，Kumar和Sastry 2017年；Wang等人2020年）。L1损失同样处理所有像素；因此，我们将像素强度ω应用到L1损失以区分相对显著的像素，并排除邻近明确或精细边缘的噪声像素，如下所示:

 （11）

为了合并上述局部和全局结构强度，我们将API损失函数组合为:

 （12）

在组合损失函数的基础上，利用地面真值G、DSi∈{0，1，2}三种深度监督，三种监督DSE的集合，以及由MEAM获得的显式边缘E，对最终损失进行了优化，结果如下：

 （13）

# 算法结果展示与分析

## 定性分析

首先，我们从视觉上直观的来分析结果，其中ours是我们的算法，与其对比的算法分别是：CVPR\_DEPTH、CVPR\_RGB、U2NET。

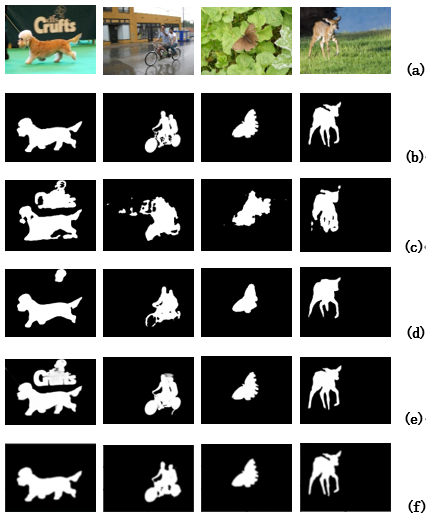


图7： DUTS-TE数据集：(a)原图；(b)理想maks；(c)CVPR-DEPTH；(d)CVPR-RGB; (e)U2NET；(f)ours

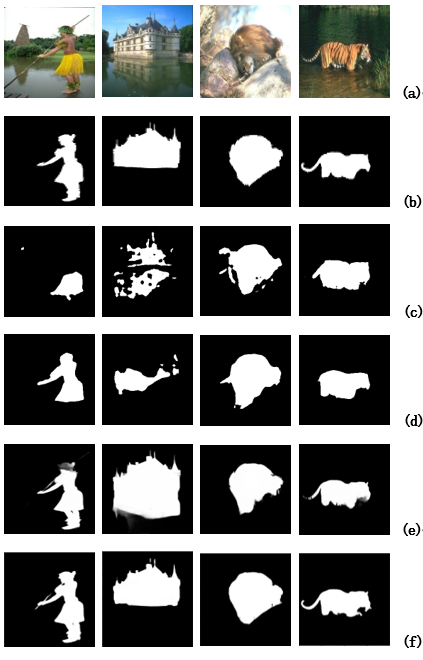


图8：ECSSD数据集：(a)原图；(b)理想maks；(c)CVPR-DEPTH；(d)CVPR-RGB; (e)U2NET；(f)ours

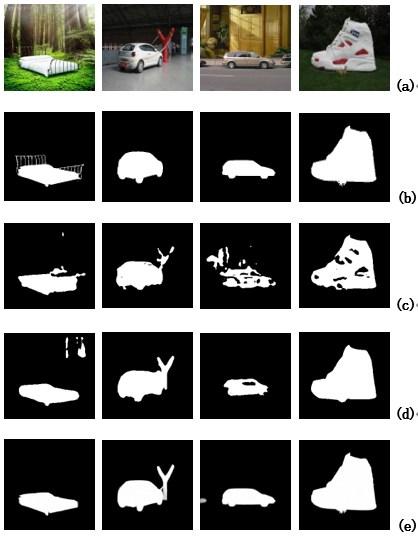


图9：DUT-OMRON数据集：(a)原图；(b)理想maks；(c)CVPR-DEPTH；(d)CVPR-RGB; (e)U2NET；(f)ours

由定性的实验结果可以看出，我们的算法结果和U2NET的算法结果在实现效果上是最好的，但是两种算法在实现显著性目标检测目标存在细边缘的时候，所检测的结果会忽略这一部分的突出目标，而将其视作是背景的部分。其次，在一些显著物体检测时候，会把一些颜色突出的部分当作目标检测出来，同时我们的算法在边缘细节的处理上还不是很好，不能很好的处理存在细边缘的显著性图片。相比之下，在显著性目标检测上，我们的算法所达到的准确性还是比较高的，都能准确地找到数据集中的显著性目标，但是在边缘细节的处理上，我们的算法所达到的效果没有U2NET算法达到的的效果好。我们分析其中的原因，是由于在我们的算法中，我们使用了屏蔽边缘注意力模块以及自适应像素强度损失函数对具有明显边缘信息的显著性目标的像素进行了权重划分，一定程度上弱化了边缘信息，但正是因为我们使用这样的方法，我们的算法性能相比于其它算法具有了明显的优势。

## 基于六种评价指标分析

我们一共使用六个评价指标来评估结果的好坏：

Maxf（max F-measure 越大越好）、Fm（F-measure 越大越好）、MAE（Mean Absolute Error越小越好）、W-Fmeasure（越大越好）、Sα（S-measure越大越好）、Eξ（E-measure越大越好）。

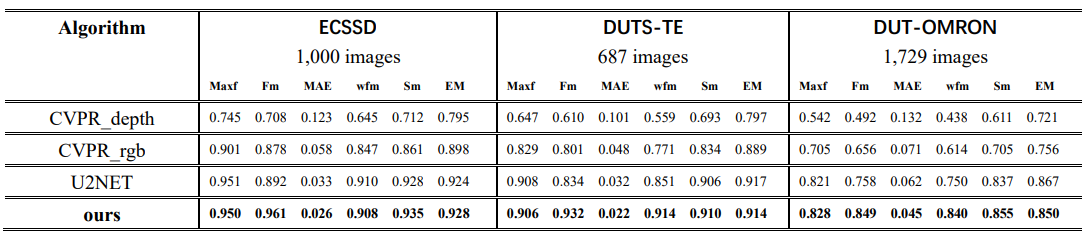


图 10 不同算法与数据集间的横向比较

对于不同数据集，我们可以发现，我们小组所参考论文的算法所达到的实验效果具有明显的竞争优势，在三个基准数据集上的表现都是非常不错的，但是与u2net算法对比，有的指标还是没有此算法所达到的效果好，主要是由于我们的算法加入了自适应像素强度损失函数和屏蔽边缘注意力模块来处理一些特殊像素，导致对于边缘的处理不够精细，没有u2net所达到的边缘处理效果好。总体来说，我们所复现的算法在三个基准数据集上具有明显的竞争力，具有一定的参考价值。

接下来，我们又从PR曲线的角度进行了分析。

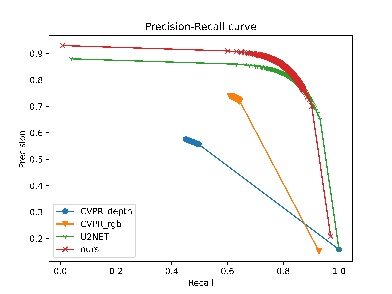
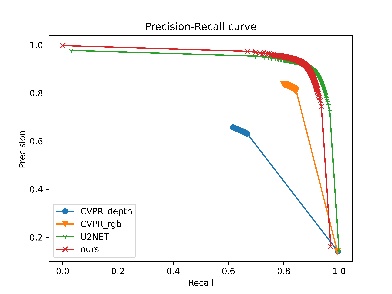
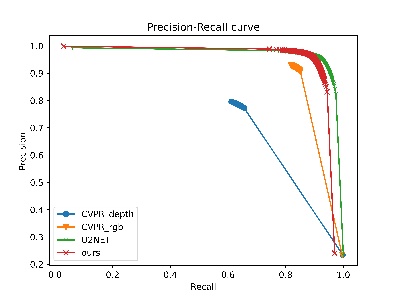


图 10 ECSSD、DUTS-TE、DUT-OMRON数据集下的各算法的PR曲线

实验结果分析：PR曲线，我们通常可以看P = R时的值来判断结果的好坏，显然有ours(TRANCER)表现最好。

# 小组成员贡献

