用于同时检测显著目标、边缘和骨架的动态特征集成方法

刘姜江, 侯淇彬, 和程明明

摘要—显著目标分割、边缘检测和骨架提取是三个对比性的低级像素视觉问题，现有的工作主要集中在为每个单独的任务设计特定的方法。然而，为每个任务存储预训练过的模型并依次执行多个不同的任务不方便而且效率低。有一些方法可以联合解决特定的相关任务，但需要同时有不同类型注释的数据集。

在本文中，我们首先展示了这些任务的一些相似之处，然后论证了如何利用它们来开发可以进行完整训练的统一框架。特别是，我们引入了一个选择性的集成模块，它允许每个任务根据自己的特性从共享主干网络中动态选择不同级别的特征。此外，我们设计了一个任务自适应注意模块，旨在根据图像内容的先验知识为不同的任务智能地分配信息。为了评估我们提出的网络在这些任务上的性能，我们在多个有代表性的数据集上进行了详尽的实验。我们将说明，尽管这些任务自然有很大不同，但我们的网络可以在这所有的任务上很好地工作，甚至比当前最好的单一目标方法的性能更好。此外，我们还进行了充分的消融分析，以完全理解所提出的框架的设计原则。

***Index Terms*—显著目标分割、边缘检测、骨架提取、联合学习**

I. 介绍

着移动设备的迅速普及，越来越多基于深度学习的随计算机视觉应用已经从计算机平台移植到移动平台。得益于其类别不可知的特性，许多低级别计算机视觉任务，成为了移动设备的基本组件。例如，在使用智能手机拍照时，许多支持任务在后台运行，以帮助用户获得更好的图片并提供实时效果预览。单摄像头智能手机通常应用显著目标分割任务来模拟需要深度信息 [1], [2] 的背景虚化效果。为了帮助用户拍摄具有更多视觉愉悦成分的照片，采用边缘检测任务来获得结构信息 [3], [4]。而骨架提取任务通过动作示意和指导用户摆更有趣的姿势，在支持拍照上起着重要的作用 [5]。然而，由于移动设备的存储和计算资源有限，为每个不同的

J.J. Liu, and M.M. Cheng are with College of Computer Science, Nankai University. M.M. Cheng is the corresponding author (cmm@nankai.edu.cn).

Q. Hou is with National University of Singapore.

The source code of this paper is publicly available on our project page: http://mmcheng.net/dfi/.

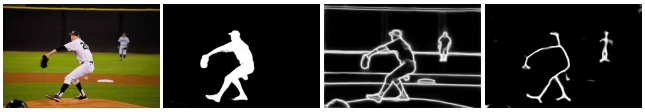


Image Saliency Edge Skeleton

图 1. 当同时学习显著性、边缘和骨架时，一个信息可能发生冲突的例子。后面的人不显著，但有骨架，对于边缘检测，它需要检测所有可能的边缘区域，无论是显著的还是属于骨架的。以上所有的预测都是通过我们的方法得到的。

应用存储预训练过的模型并依次执行多个不同的任务是不方便且效率低的。

一个可行的解决方案是在单个模型中执行上述任务，但存在两个主要挑战。一是如何同时学习不同的任务，二是如何解决不同任务的特征域和优化目标的分歧。大多数先前的工作 [6]–[9] 通过观察不同任务所具有的特性并为每个任务手动设计专门的网络结构来解决第一个挑战。他们假设联合学习的所有任务是互补的，而一些任务是辅助的 (例如，利用额外的边缘信息来帮助显著目标检测任务在边缘区域进行更精确的分割)。通常辅助任务的性能会被牺牲和忽略。但是当面临第二个挑战时，所要解决的任务是对比的, 如 Fig. 1, 所示，直接应用这些方法往往会失败。如 Tab. III, 中第三行所示, 与其他两个任务联合训练时，骨骼提取的性能严重受损。

以前工作的设计标准通常是面向任务的和特定的，这极大地限制了它们对其他任务的适用性 [6]。从网络架构的角度来看，尽管有三个不同的任务，它们都需要多级别的特征，然而程度不同。显著目标分割需要提取均匀区域的能力，因此更依赖于高级特征 [1]。边缘检测旨在检测精确的边界，因此需要更多的低层次特征来锐化由较深层生成的粗糙边缘图 [10], [11]。骨架提取 [12], [13] 更喜欢低、中和高层次信息的适当组合，以检测多尺度 (厚或薄) 的骨架。因此，自然有一个问题，是否有可能设计一种架构，能够将这三个对比的、低级的视觉任务合并成一个统一但完整的可训练网络，而不损失每个任务的性能。

考虑到每项任务的不同特点，我们提出了一个新的、统一的框架来解决上述挑战。具体来说，我们的网

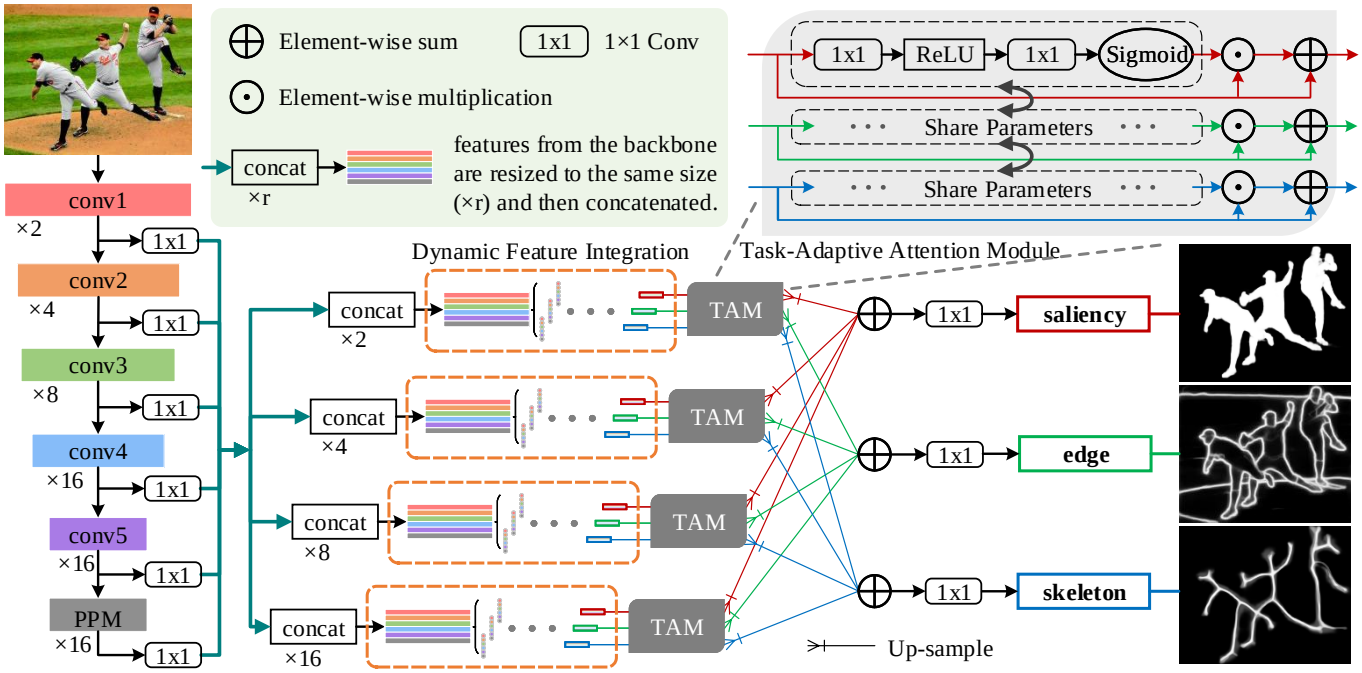


图 2. 我们所提出的的方法的整体流程 (彩色图视觉效果最佳)。

络包含一个共享的主干网络和三个相同设计的任务分支，如 Fig. 2所示。为了便于每个任务分支在主干网络的不同级别中自动选择适当的特征，我们引入一种动态特征集成策略，它能够以完整的学习方式动态选择有利的特征。这种动态策略可以极大地简化架构构建的过程，并促进主干网络适当地调整其参数来解决多个问题。然后采用任务自适应注意模块，以分离-聚集的方式实现不同任务分支之间的信息交换。通过结合以前独立的分支，我们可以避免网络的不对称优化。我们的方法很容易使用，并且可以在单个 GPU 上进行完整的训练。在不牺牲性能的情况下，当处理 300×400 的图像并同时执行这三个任务时，它可以达到 40 FPS 的速度。

为了评估所提出的架构的性能，我们将其与三个任务目前最好的方法进行了比较。实验结果表明，在多个广泛使用的基准上，我们的方法都优于现有的单一目的方法。具体来说，对于显著目标分割，与以前的最好的工作相比，我们在六个流行的数据集上进行测试，我们的方法在 F-measure 上平均具有 1.2% 的性能增益。对于骨架提取，我们还将在 SK-LARGE 数据集 [14] 上最好的结果在 F-measure 方面提高了 1.9%。此外，为了让读者更好地理解所提出的方法，我们对所提出的架构的不同组件进行了大量的消融实验。

综上所述，本文的贡献可以概括为: (i) 我们设计了一种动态特征集成策略，根据每个输入和任务自动探索特征组合，并在一个完整的统一框架中同时解决三个对比任务，运行速度为 40FPS。(ii) 我们将我们的多任务方法与每项任务的单一目的、目前最好的方法进行比较，我们的方法有着更好的性能。

II. 相关工作

A. 相关的二元任务

对于显著目标分割，传统方法主要基于手工特征 [15]–[21]. 随着 CNNs 的普及，很多方法 [22]–[27] 开始使用 CNNs 提取特征。它们中的一些方法 [28]–[33] 结合了迭代和循环学习的思想来改进预测。也有工作利用融合更丰富的特征 [34]–[43]、引入注意机制 [44]–[46] 、使用多个阶段以逐阶段的方式学习预测 [30], [47], [48] 或添加更多的监督以获得更尖锐边缘的预测 [49]–[58] 等方法解决问题。对于边缘检测，早期的工作 [59]–[61] 主要依赖于各种梯度算子。后来的工作 [62]–[64] 进一步采用了手工设计的特征。最近，基于 CNN 的方法通过 patch-wise [65]–[68] 或 pixel-wise 预测方式 [69]–[73] 使用完全卷积网络 (fully-convolutional networks) 来解决这个问题。对于骨架提取，较早的方法 [74]–[76] 主要依靠自然图像的梯度强度图来提取骨架。后来，基于学习的方法 [77]–[80] 将骨架提取视为像素分类问题或超像素聚类问题。最近的方法 [5], [12], [13], [81] 分层地考虑这个问题，设计了强大的网络结构。与上述所有方法不同，我们的方法在一个统一的框架内同时解决三个任务，而不是用一个单独的网络来学习每个任务。

B. 多任务学习

多任务学习 (MTL) 在机器学习领域有着悠久的历史 [82]–[85]。最近，已经提出了许多基于 CNN 的 MTL 方法，其中大多数集中在网络架构的设计 [6], [86]–[88]，或平衡不同任务的重要性的损失函数 [89], [90]，或两者兼有 [91]。不同的工作也解决了不同的任务组合，包括：多领域图像分类 [92]；目标识别、定位和检测 [93]–[95]；姿势估计和动作识别 [96]–[98]；语义类别、表面法线和深度预测 [86], [89], [91], [99]–[101]。然而，这些方法中的大多数集中于特定的相关任务，这些任务需要同时支持具有不同类型注释的数据集。与上述方法不同，我们的目标是将动态特征集成的思想融入到架构设计中。这允许我们的方法使用多个独立数据集的训练数据一起学习多个任务。此外，不同于以前的方法 [6], [7] ，这些方法固定了特征集成到网络结构中的策略，而我们的方法可以调整网络连接来动态地选择特征，从而促进多任务训练。

C. 门控机制

门控机制首先被引入到自然语言和语音处理领域。最近的工作将其应用到各种计算机视觉任务中，并证明它的有效性。对于语义分割任务，Qi [102] 提出使用层间的记忆门来学习每个单独像素的自定义比例的特征表示。Takikawa*et al.* [103] 使用经典流中的高级激活来门控形状流中的低级激活，这有效地去除了噪声信息。Ding*et al.* [104] 提出了一种门控和的方案，为每个空间位置选择性地聚集多尺度特征。Cheng *et al.* [105] 利用 RGB-D 信息，设计了一个门控融合层来结合 RGB 和深度特征。Zhu*et al.* [106] 和 Li*et al.* [107] 解决了目标检测问题，并使用选通技术来选择锚框特征。在图像分类任务中，Chen*et al.* [108] 提出了一种 gater 网络来从主干网络中选择过滤器，而 [109], [110] 设计了一种软选通机制，允许每个神经元自适应地调整其感受野的大小。Hua 等人 *et al.* [111] 采用门控机制，切断那些不太重要的通道，实现网络中的枝剪。与上述方法不同，我们利用门控机制同时解决三个对比任务。此外，与按像素、通道或图层选择特征不同，我们分阶段从主干网络中选择特征。

III. 方法

在本节中，我们计划令网络根据每个任务的偏好和每个输入的内容动态地选择不同阶段的特征，而不是试图手动地设计一个可能适用于所有三个任务的架构，如 Sec. I所述。

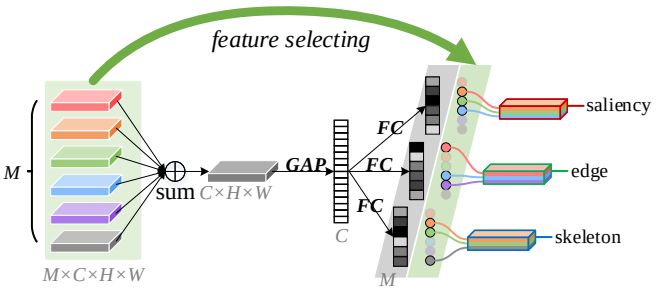


图 3. DFIM 的详细说明。它将从主干中提取的特征集作为输入，首先将其大小调整为相同的大小。然后为每个任务动态选择不同阶段的特征。

A. 总体流程

我们在一个可以进行完整训练的统一网络中，多个独立的数据集上有着三个不同的任务（即，用于显著性的 DUTS [112]，用于边缘的 BSDS 500 [64] 和 VOC Context [113] ，用于骨架的 SK-LARGE [12] 或 SYMPASCAL [13]）。所有数据集都能直接使用，就和现有的、为每个任务所提出的单一目标的方法对这些数据集的使用方法一样，无需额外处理。

Fig. 2显示了所提出的框架的总体流程。我们使用 ResNet-50 [114] 网络作为特征提取器。我们将 `conv_1` 输出的特征图作为 S_1，并将 `conv2_3`, `conv3_4`, `conv4_6`, 和 `conv5_3` 的输出分别作为 S_2 至 S_5。我们将 `conv5` 中 3×3 卷积层的扩展比率设置为 2，就像在像素级预测任务中所做的那样。此外，我们在 ResNet50 的顶部添加了一个金字塔池化模块 (PPM) [115]，以捕获更多的全局信息，就如 [8], [48] 中所做。输出表示为 S_6。不同于大多数以前的单目标方法中所做的，在网络结构中手动固定特征集成策略，如在，不同输出下采样率（ Fig. 2中橙色虚线圆角矩形）的一系列动态特征集成模块 (DFIMs) 被安排来动态地、分别地为三个任务结合从主干网络 (即 $\{S_i\}$，其 $1 \le i \le M$ 且 $M = 6$) 提取的特征。然后在每个 DFIM 之后连接一个任务自适应注意模块 (TAM)，以智能地在任务之间分配信息，防止网络有偏向地优化。最后，由每个任务的 TAMs 输出的相应特征图被上采样并求和，然后分别再连接 1×1 卷积层用于最终预测。

B. 动态特征集成

在许多以前的多任务方法 [6], [86], [89], [91] 中已经提到，不同任务所需的特征差异很大。而且大多数都需要在单个数据集内进行多种标注，这是很难获得的。不同的是，我们利用来自多个独立数据集的不同任务的训练数据，并且这更有可能满足不同任务所需的特

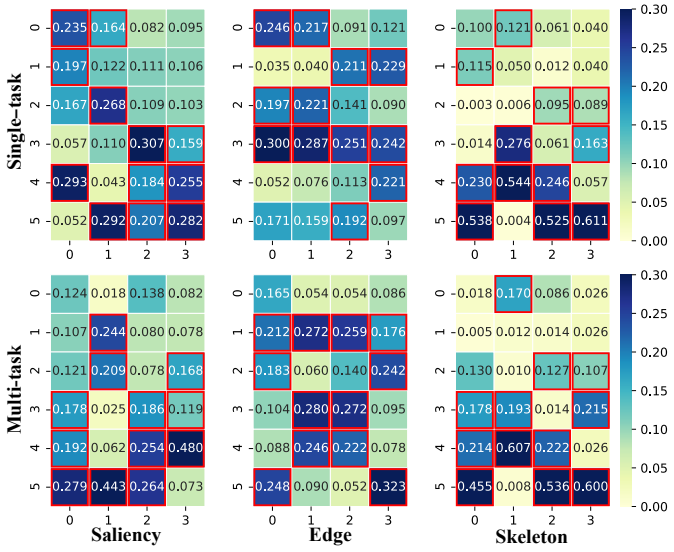


图 4. 每个阶段特征的权重由每个 DFIM 选择。在每个子图中，行表示 DFIMs 的索引，列表示特征的阶段。每个 DFIM 只保留上半部分（红色矩形）。

征相冲突的情况，如 Fig. 1所示。为了解决这个问题，我们提出了 DFIM，它在训练和测试期间根据每个任务和输入动态地调整特征集成策略。现有方法根据不同任务特征的人工观察从主干网络中集成特定级别特征，与之相比 DFIM 可以学习特征集成策略。

具体来说，我们将从主干网络提取的特征集 $\{S_i\}$ 作为每个 DFIM 的输入。并且在网络定义期间确定每个 DFIM 所需的输出下采样率 $\times r$ 。如 Fig. 3所示，分别通过 1×1 卷积层和双线性插值，我们首先将具有相同数量的信道（即，$C\times$）和下采样率（即，$\times r$）的特征 $\{S_i\}$ 记为 $\{S_i^r\}$。为了使 DFIM 能够查看所有要选择的功能，我们对 $\{S_i^r\}$ 求和，并在其后连接一个全局平均池化（GAP）层，以创建一个紧凑的全局特征（$C\times$），就如 SENet [116] 所做。对于每个任务 $t \in \{saliency, edge, skeleton\}$，我们使用独立的全连通 (FC) 层将 $C\times$ 特征映射到 $M\times$ 通道，然后应用 softmax 算子将 $M\times$ 特征转换为概率的形式 $\{p_i^{r,t}\}$ $(1 \le i \le M)$，该概率可用作选择特征的指标。由于并非来自主干网络的每个阶段的特征总是有帮助的，这不同于现有的低级视觉方法 [3], [5], [10] ，保持 $\{S_i^r\}$ 的紧密连接，我们仅保持一半的连接记为 $\{S_i^{r,t}\}$ $(1 \le i \le M)$:

$$S_i^{r,t} = \begin{cases} p_i^{r,t} * S_i^r, & \text{if } p_i^{r,t} \ge median(\{p_i^{r,t}\}) \\ 0, & \text{else,} \end{cases} \tag{1}$$

其中，$median(\cdot)$ 表示取中值，且 $1 \le i \le M$。因此，任务 t 的下采样率为 $\times r$ 的 DFIM 输出可通过下式获得

$$D^{r,t} = \sum_i S_i^{r,t}. \tag{2}$$

对这种设计选择的详细分析在 Sec. V-B中有所描述。

通过排列一系列各种下采样率的 DFIMs，可以得到动态结合的特征图 $\{D^{r,t}\}(r \in \{2,4,8,16\}, t \in \{saliency, edge, skeleton\})$，如 Fig. 2所示。由于特征集成策略仅取决于输入和任务类型，网络能够以完整的方式在更宽、更灵活的特征组合空间内学习每个输入和任务的集成策略。

C. 任务自适应注意

当我们利用来自多个独立数据集的训练数据时，不能忽略域转移 [82], [117] 问题。如何有效地合并来自不同数据集的信息对于维护所有任务的整体性能是不可或缺的。如 Fig. 4的第一行（单任务）所示，不同任务所偏好的特征的级别差异很大。如果我们直接使用由 DFIMs 生成的任务特定特征图 $\{D^{r,t}\}(r \in \{2,4,8,16\})$ 来预测每个任务，则一些任务到网络共享部分的梯度可能明显偏离其他任务，这将导致优化方向偏转到局部最小值并引起欠拟合。

为此，我们打算在来自主干网络的共享特征被动态集成并为每个任务定制之后，让网络具有智能地为不同任务分配信息的能力。如 Fig. 2的右上角所示，来自 DFIM 的输出特征图 $\{D^{r,t}\}(t \in \{saliency, edge, skeleton\})$ 以 $\times r$ 的下采样率被进一步传送到 TAM。

在每个 TAM 模块中，我们首先将输入特征图 $D^{r,t} \in \mathbb{R}^{C\times H\times W}$ 喂入到 1×1 卷积层 $(f_1^{1\times1})$，以减少上采样后特征映射添加的混叠效应 (Eqn. (2))，随后是 ReLU 激活函数以引入非线性。然后采用另一个 1×1 卷积层 $(f_2^{1\times1})$ 来映射交叉通道信息。之后我们应用一个 sigmoid 层 (σ) 计算空间注意图 $A^{r,t} \in \mathbb{R}^{C\times H\times W}$:

$$A^{r,t} = \sigma(f_2^{1\times1}(ReLU(f_1^{1\times1}(D^{r,t})))), \tag{3}$$

其中 $f_1^{1\times1}$ 和 $f_2^{1\times1}$ 中的参数在任务间共享。有了输入特征图和它的注意图，就可通过以下方式获得最终的 TAM 输出特征图:

$$T^{r,t} = D^{r,t} \odot (1 + A^{r,t}), \tag{4}$$

其中 $\odot$ 表示逐元素乘法。$D^{r,t} \odot A^{r,t}$ 作为输入特征图的残差。

为了交换信息，我们在任务间共享可学习的参数。与直接使用每个任务的输出相比，包含所有任务关系

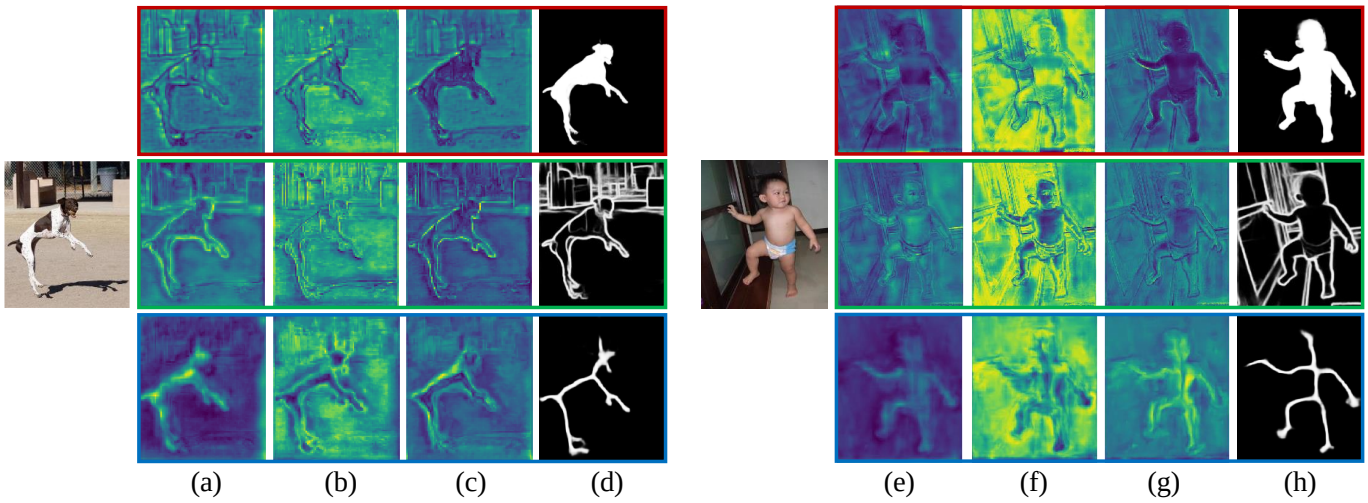


图 5. 在 TAM 附近可视化特征图。(a,e)TAM 之前；(b,f)TAM 中的注意图；(c,g)TAM 之后；(d,h) 预测结果。可以看出，TAM 可以自适应地和相应地为每个任务定制特征图。从上到下依次为: 更清晰的显著部分，更锐利的边缘和更强健的骨架。

的额外建模使 DFIM 能够通过同时考虑输入内容和所有任务的特性，自适应地调整每个任务对共享主干网络的影响。即使在为每个任务分离和定制了特征图之后，TAM 也能强制跨任务交换信息。这与以前的方法 [6], [7], [91] 大不相同，以前的方法保证不同任务的分支相互独立，直到结束。

为了有更好的理解，我们在 Fig. 5中围绕 TAM 可视化中间特征图。可以看到，第一行，对于显著目标分割，在 TAM (a, e) 之前，很难从背景中分辨狗 (小孩)。TAM (b, f) 中学习的注意图有效地擦除了背景的激活。而在 TAM (c, g) 之后，狗 (小孩) 就清楚地凸显出来了。在第二行中，对于边缘检测，TAM (c, g) 之后的特征图与 TAM (a, e) 之前的厚和模糊的激活相比，在可能存在边缘的区域中具有明显更薄和更尖锐的激活。骨架提取任务中也可以观察到类似的现象。如最后一行所示，经过 TAM 后狗 (小孩) 的骨架变得更强壮更清晰。所有上述讨论都验证了 TAM 在更好地为不同任务分配信息方面的显著效果。

IV. 实验设计

在本节中，我们描述了实验设计，包括所提出的网络的实现细节、所使用的数据集、训练过程和三项任务的评估指标。

实现细节. 我们基于 PyTorch[1]实现了所提出的方法。所有实验都是在一个工作站上进行的，该工作站配有一个 Intel Xeon 12 核 CPU (3.6GHz)、64GB RAM 和一个 NVIDIA RTX-2080Ti GPU。我们使用 Adam [118] 优化器，初始学习率为 5e-5，权重衰减为 5e-4。我们

[1]https://pytorch.org

表 I
我们用于训练和测试的数据集。

Task	Training	#Images	Testing	#Images
Saliency	DUTS-TR [112]	10553	ECSSD [121], PASCAL-S [122], DUT-OMRON [123], SOD [124], HKU-IS [22], DUTS-TE [112]	1000, 850, 5166, 300, 1447, 5019
Edge	BSDS500 [64] & VOC Context [113]	300 + 10103	BSDS500 [64]	200
Skeleton	SK-LARGE [14]	746	SK-LARGE [14]	745
	SYM-PASCAL [13]	648	SYM-PASCAL [13]	788

的网络一共训练了 12 代，9 代后学习率除以 10。我们网络主干 (即 ResNet50 [114]) 的参数是用 ImageNet [119] 预训练过的模型初始化的，而其他所有参数都是随机初始化的。除主干外，在每个卷积层之后应用组归一化 [120]。我们网络中所有参数的优化配置都是相同的，除了主干的批处理归一化层的参数，它们在训练和测试期间是被冻结。

数据集. 我们为不同的任务使用单独的数据集，每个数据集只有一种注释。Tab. I中列出了详细的配置。所有数据集的使用方法，都与为每个任务 [3], [5], [37] 提出的现有单一目标方法对数据集的使用一样，没有额外的预处理。

训练步骤. 为了以完整的方式在三个单独的数据集上联合解决三个不同的任务，对于每次迭代，我们分别为三个任务中的每一个随机采样一个图像-正确标注对。然后，依次将三个图像-正确标注对中的每一个传递到网络，并计算相应的损失。最后，我们简单地通过网络将三个计算的损失相加，然后进行优化。所有其他训练

步骤与典型的单一目标方法相同。

损失函数. 我们定义这三个任务的损失函数与大多数以前的单目标方法相同。我们使用标准二元交叉熵损失用于显著目标分割 [1], [37] ，使用平衡二元交叉熵损失 [3], [5], [10] 用于边缘检测和骨架提取。我们使用的损失函数的详细公式如下。给定图像的预测图 $\hat{Y}$ 及其对应的正确标注图 Y，对于所有像素 (i,j)，我们将标准二进制交叉熵损失计算为:

$$\begin{aligned}\mathcal{L}_s(\hat{Y},Y) = -\sum_{i,j}[&Y(i,j)\cdot\log\hat{Y}(i,j)+\\&(1-Y(i,j))\cdot\log(1-\hat{Y}(i,j))],\end{aligned} \tag{5}$$

并且平衡的二元交叉熵损失为:

$$\begin{aligned}\mathcal{L}_b(\hat{Y},Y) = -\sum_{i,j}[&\beta\cdot Y(i,j)\cdot\log\hat{Y}(i,j)+\\&(1-\beta)\cdot(1-Y(i,j))\cdot\log(1-\hat{Y}(i,j))],\end{aligned} \tag{6}$$

其中 $\beta = |Y^-|/|Y^+ + Y^-|$ 而 Y^+ 和 Y^- 分别指前景和背景像素。

- 显著目标分割:

$$\mathcal{L}_{sal}(\hat{Y}_{sal},Y_{sal}) = \mathcal{L}_s(\hat{Y}_{sal},Y_{sal}). \tag{7}$$

- 边缘检测:

$$\mathcal{L}_{edg}(\hat{Y}_{edg},Y_{edg}) = \mathcal{L}_b(\hat{Y}_{edg},Y_{edg}), \tag{8}$$

  其中 $\mathcal{L}_b$ 的 β 中的 Y^+ 指的是边缘像素，而 Y^- 指的是非边缘像素。
- 骨架提取:

$$\mathcal{L}_{sk}(\hat{Y}_{sk},Y_{sk}) = \mathcal{L}_b(\hat{Y}_{sk},Y_{sk}), \tag{9}$$

  其中 $\mathcal{L}_b$ 的 β 中的 Y^+ 指骨架像素，而 Y^- 指非骨架像素。

总损失计算为三项任务损失的简单总和，它们同等重要:

$$\mathcal{L} = \mathcal{L}_{sal}(\hat{Y}_{sal},Y_{sal}) + \mathcal{L}_{edg}(\hat{Y}_{edg},Y_{edg}) + \mathcal{L}_{sk}(\hat{Y}_{sk},Y_{sk}). \tag{10}$$

评估标准. 对于显著目标分割，我们使用精确度-召回率 (PR) 曲线、F-measure 得分 (F_β)、平均绝对误差 (MAE) 和 S-measure (S_m) [125] 进行评估。F-measure 分数由精确度和召回率的加权调和平均值计算得出，是对性能的总体衡量:

$$F_\beta = \frac{(1+\beta^2)\times Precision\times Recall}{\beta^2\times Precision + Recall}, \tag{11}$$

表 II
所提出网络的参数构成。可以看出，特征提取器 (RESNET-50 & PPM) 和共享部分占大多数。

Total: 29.57M						
Shared: 27.01M (91.34%)				Specific: 2.56M (8.66%)		
ResNet-50	PPM	DFIMs	TAMs	Saliency	Edge	Skeleton
23.46M	1.31M	1.42M	0.83M	0.85M	0.85M	0.85M
79.34%	4.43%	4.80%	2.81%	2.87%	2.87%	2.87%

其中 β^2 被设置为 0.3，如在先前的工作中所做的那样，以比召回率更偏重精确度。MAE 分数测量二元正确标注 G 和预测显著图 P 之间的平均像素绝对差异。

$$MAE = \frac{1}{W\times H}\sum_{x=1}^{W}\sum_{y=1}^{H}|P(x,y)-G(x,y)|, \tag{12}$$

其中 W 和 H 分别表示 P 的宽度和高度。通过同时考虑目标感知 (S_o) 和区域感知 (S_r) 结构的相似性，S-measure 分数评估结构相似性:

$$S_m = \alpha\times S_o + (1-\alpha)\times S_r, \tag{13}$$

其中 α 根据经验设定为 0.5。

对于 边缘检测, 在评估之前，我们应用标准的非最大抑制 (NMS) 算法 [126] 来获得细化的边缘。为了产生二元边缘图，有两种选择来设置阈值。一种是对数据集中的所有图像使用固定的阈值，从而在整个数据集上提供最佳的整体性能。我们称之为最优数据集尺度 (ODS)。另一种是为每幅图像选择一个最佳阈值，称为最优图像尺度 (OIS)。对于 ODS 和 OIS，我们都给出了 F-measure 分数:

$$F_m = \frac{2\times Precision\times Recall}{Precision + Recall}. \tag{14}$$

对于 骨架提取, 我们遵循 [77] 中的评估方案。我们给出了精确度-召回率 (PR) 曲线和最大 F-measure 分数 (F_m)。在评估之前，预测的骨架图通常被 NMS 细化。为了获得 PR 曲线，给定一个 NMS 细化的骨架图，我们首先将其阈值化为一个二值图，然后将其与相应的正确标注图进行匹配。在匹配期间，在预测的正像素和正确标注的骨架像素之间允许微小的定位误差。通过对预测的骨架图应用不同的阈值，得到一系列的精确度和召回率对来绘制 PR 曲线。最大 F-measure 是在整个数据集的最佳阈值下使用 Eqn. (14) 获得的。

V. 消融研究

在本小节中，我们首先分析所提出的模型的参数组成。然后我们通过在单任务和多任务环境下进行实

表 III
在五个广泛使用的数据集上的显著目标分割、边缘检测和骨架提取的定量结果。“单一任务”是指直接应用我们的方法，同时只执行单一任务。每一栏中的最佳结果以**粗体**突出显示。

No.	DFIM	TAM	Saliency									Edge		Skeleton
			PASCAL-S [122]			DUT-OMRON [123]			DUTS-TE [112]			BSDS 500 [64]		SK-LARGE [14]
			$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	ODS $\uparrow$	OIS $\uparrow$	$F_m\uparrow$
Our Method (Single-Task)														
1	sparse	w/o	0.860	0.075	0.849	0.811	0.059	0.835	0.875	0.042	0.878	0.815	0.831	0.749
2	sparse	independent	0.859	0.081	0.849	0.817	0.060	0.835	0.880	0.045	0.878	0.812	0.826	0.746
Our Method (Multi-Task)														
3	identity	w/o	0.877	**0.062**	**0.865**	0.818	0.056	0.836	0.885	**0.038**	0.886	0.811	0.828	0.708
4	dense	w/o	0.872	0.064	0.859	0.813	0.056	0.833	0.877	0.039	0.881	0.810	0.825	0.740
5	sparse	w/o	0.874	0.064	0.862	0.817	0.056	**0.842**	0.884	**0.038**	**0.887**	0.818	0.834	0.744
6	sparse	independent	0.873	0.065	0.861	0.815	0.057	0.836	0.879	0.039	0.883	0.815	0.832	**0.753**
7	sparse	share	**0.880**	0.065	**0.865**	**0.829**	**0.055**	0.839	**0.888**	**0.038**	**0.887**	**0.819**	**0.836**	0.751
Other Methods (Multi-Task)														
8	UberNet_{17} [6]		0.823	-	-	-	-	-	-	-	-	0.785	0.805	-
9	MLMS_{19} [7]		0.853	0.074	0.844	0.793	0.063	0.809	0.854	0.048	0.862	0.769	0.780	-

验来研究所提出的 DFIM 算法的有效性。最后我们展示了 TAM 的效果，它具有更好的整体收敛性和性能。

A. 参数的组成

我们在 Tab. II中列出了网络参数的组成。可以看出，91.34% 的参数在任务间共享，其中特征提取器 (ResNet-50 & PPM) 占 91.71%。而 DFIMs 和 TAMs 的共享部分只引入了 2.25M (8.33%) 个参数。每个任务分别拥有 0.85M (2.87%) 个特定任务的参数。参数的极化组合证明了该方法的有效性和高效性。通过利用从主干提取的共享特征并自适应地重组它们，可以节省更多的参数和空间。同时，网络本身设计特征整合策略，所需的人工交互较少。

B. 动态特征集成

动态特征集成的有效性. 如 Tab. III的第一行所示，直接应用我们的方法，同时只执行单个任务，在显著目标检测和边缘检测任务方面，可以获得能与目前最好的方法可比较的结果。在骨架提取任务 (1.7%) 上可以观察到更大的提升。这表明所提出的 DFIM 能够根据所要解决的目标任务的特征来调整其特征选择策略。与以前的方法中通常手工为不同的任务设计特定网络结构不同，DFIM 所需的人工交互较少。

三项任务共同学习时 (Tab. III中第 5 行), 在三个数据集上，显著目标分割任务的性能几乎在所有方面都有明显的提高。这与以前的研究一致，即边缘信息可以帮助显著目标分割任务在边缘区域进行更精确的分割。边缘检测任务的性能也提高了，这表明显著目标的边缘也可以提供有用的引导信号。骨架提取任务的性能仅略微下降。

为了对联合训练这三个任务的难度进行数值估计，我们通过移除 DFIM 的动态特征选择过程来建立基线 (在 Tab. III中的第 3 行, 记为 “identity”)。这意味着从主干中提取的特征再求和之后的所有操作都将被删除，如 Fig. 3所示。这也等同于将 Eqn. (2) 替换为下式:

$$D^{r,t} = \sum_i S_i^r. \tag{15}$$

通过将 “identity” 版本与所提出的 “sparse” 特征选择版本 (第 5 行) 进行比较，我们可以观察到边缘检测和骨架提取任务上的明显下降，分别为 0.7% 和 3.6%。这些现象表明，简单地融合所有级别的特征会破坏边缘和骨架的检测。当所涉及的任务具有不同的优化目标并从不同的数据集获取训练样本时，很难手工设计网络结构。类似的情况在以前的工作 [6], [7] 中也有发生，其中部分任务的性能在联合解决不同任务时会显著下降，如 Tab. III的最后两行所示。但是对于 DFIM 来说，通过让网络本身动态地和相应地集成特征，所有三个任务的执行相当于分别训练每个任务。

动态学习整合策略. 为了更好地理解我们提出的方法已经学习了什么特征集成策略，我们从 DUTS(显著性)、BSDS500(边缘) 和 SK-LARGE(骨架) 的每个测试集中随机选择 100 幅图像来形成包含 300 幅图像的

表 IV
我们的方法对 DFIMS 不同下采样率组合的消融分析。每一栏中的最佳结果以**粗体**突出显示。

Down-sampling Rates	Saliency						Edge		Skeleton
	DUT-OMRON			DUTS-TE			BSDS 500		SK-LAR
	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	ODS $\uparrow$	OIS $\uparrow$	$F_m\uparrow$
2,4,8	0.814	0.057	0.839	0.879	**0.038**	0.885	0.815	0.829	0.742
4,8,16	0.809	0.057	0.837	0.880	**0.038**	0.884	0.814	0.829	**0.745**
2,4,8,16	**0.817**	**0.056**	**0.842**	**0.884**	0.038	**0.887**	**0.818**	**0.834**	0.744

集合。通过传递这些图像，我们对所有图像的 $\{p_i^{r,t}\}$ 值进行平均，这些值指示特征选择。我们得到每个 DFIM 在每个阶段为不同任务从主干中选择特征的概率，并在 Fig. 4中绘制。我们按列比较子图，会发现不同任务偏好的特征阶段会有很大不同。这可以解释为什么一个任务的具有良好性能的架构不能在其他任务上工作 [1], [3], [81]。如果我们逐行比较子图，当三个任务中的每一个以单任务方式单独训练时所选择的特征阶段也与它们以多任务方式联合训练时的选择特征阶段有很大不同。这可能是这三个任务中的任一个都被很好地研究过，但很少有文献试图共同解决它们的原因。手动设计架构很难，因为共享主干的特性现在将同时受到所有任务的影响。

稀疏或密集连接. 在 Tab. III中，我们将我们的稀疏连接网络与密集连接网络进行比较，密集连接网络的 $\{S_i\}$ $(1 \le i \le M, M = 6)$ 中的所有特征图都被保留，而不是 Eqn. (1) 中的一半。如第 4 行和第 5 行所示，密集版本几乎在所有的三个任务上的性能都更差。这可能表明，并不是来自主干的每个阶段的特征总是有帮助的 [127]。例如，对于边缘检测，更低级别的特征图对于边缘像素的精确定位是必要的 [3], [10] ，而对于骨架提取，更高级别的信息对于确定像素是否是骨架是必要的 [5], [81]。

DFIMs 的下采样率. 如 Tab. IV中所列出的，我们对 DFIMs 下采样率的组合进行了消融实验。更大范围的下采样率显示出更好的整体性能平衡，特别是在显著目标分割和边缘检测方面，这与更丰富的多尺度信息通常有所帮助的常识相一致。

C. 任务自适应注意

TAM 的有效性. 有了 DFIM，我们可以在一个统一的架构下联合训练这三个任务。但是，如 Tab. III的第 5 行所示，与单独训练 (第 1 行) 相比，骨架提取的性能下降。由于显著目标分割和边缘检测任务的注释更多地关注边缘存在的像素，这与骨架提取任务的目标不符，因此骨架提取任务的优化可能会受到影响并误导向相反的方向。有了 TAM，网络通过自适应调整每个任务传递给共享主干的梯度，能够从全局角度分配所有任务的信息。从 Tab. III中第 7 行与第 5 行的对比可以看出，我们取得了更好的整体性能。显著目标分割和边缘检测的性能稍好，而骨架提取的性能提升 0.7%。

信息交换的必要性. 为了研究 TAM 带来的提升是否是由于额外参数的引入，我们还进行了实验，将 TAM 中不同分支的参数保持不共享 (第 6 行)，以便不同的任务分支在从共享主干中选择特征后保持独立。因不共享 TAM 中的参数，额外的 1.66M 个参数被进一步引入。然而，从 Tab. III的第 6 行可以看出，即使引入更多参数，非共享版本的 TAM 的整体性能也明显不如共享版本的 TAM(第 7 行)。虽然骨架任务表现稍好，但在其他两个任务上的表现却大幅下降。这些现象表明，在每个任务分别进入各自分支之后，强制跨任务交换信息有助于所有任务的整体收敛，而简单的注意力机制却无法很好地发挥作用。这也可以从 Tab. III的前两行观察到，当每个任务被分别训练时，附加 TAM 对三个任务中的多数没有帮助，甚至产生负面影响。

VI. 与最先进方法的比较

在这一节中，我们将所提出的方法 (为了方便起见记为 DFI) 与最先进的方法在显著目标分割、边缘检测和骨架提取方面进行了比较。由于以前很少有文献联合解决这三个任务，例如 UberNet [6] (CVPR'17) 和 MLMS [7] (CVPR'19)，它们联合解决了显著目标分割和边缘检测，我们主要与这三个任务的最先进的单一目标方法进行比较，以便更好地说明。为了公平比较，对于每个任务，其他方法的预测图 (例如，显著图、边缘图、骨架图) 由作者发布的原始代码生成或由他们直接提供。除了评估边缘和骨架图之前的 NMS 过程 [3], [5], [10]，所有结果都是直接从单模型试验中获得的，不依赖于任何其他预处理或后处理工具。对于每个任务，所有的预测图都用相同的评估代码进行评估。

A. 显著目标分割

我们详尽地比较了 DFI 和现有的 16 种最先进的显著对象分割方法，包括 DCL [34], RFCN [28], MSR [35], DSS [1], NLDF [36], Amulet [37], PAGR [31], DGRL [30], MLMS [7], JDFPR [33], PAGE [50], CapSal [53], CPD [40], PiCANet [45], AFNet [51], 和 BASNet [54].

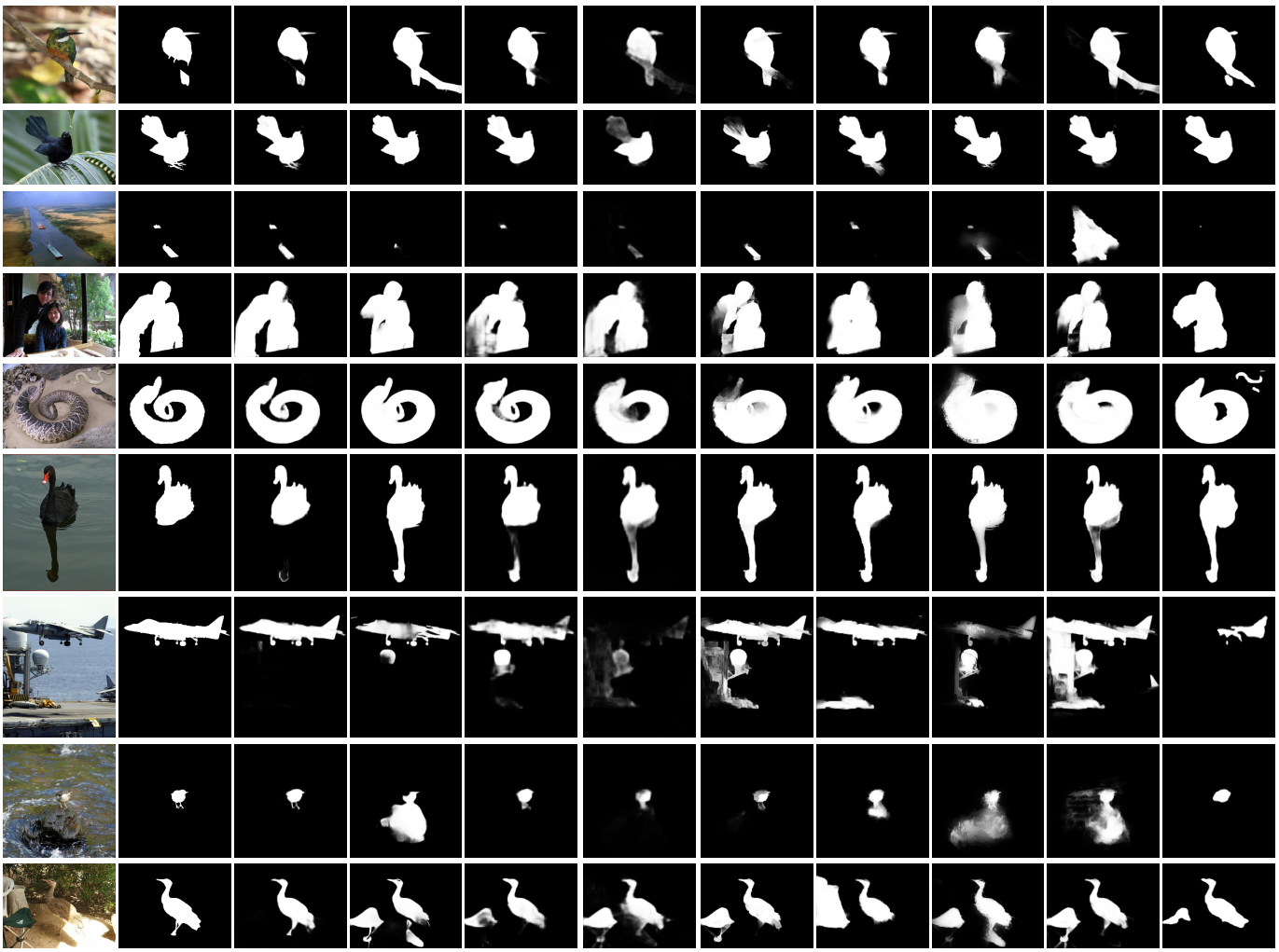


图 6. 不同显著目标分割方法的可视化比较。

F-measure, MAE 和 S-measure 分数. 在这里，我们比较了 DFI 和前面提到的方法在 F-measure，MAE 和 S-measure 方面的差异 (见 Tab. V)。可以看出，与每个数据集上的第二好的方法相比，DFI 在六个数据集上的表现优于所有这些方法，在 F-measure 和 S-measure 方面的平均提升分别为 1.2% 和 1.0%。特别是在具有挑战性的 DUTSTE 数据集上，可以观察到 F-measure 和 S-measure 有 2.1% 和 1.8% 的提升。使用 MAE 分数也可以观察到类似的结果。此外，与联合学习显著目标分割和边缘检测的 MLMS [7] 相比，DFI 在这两项任务上都有更大的改进，如 Tab. III的第 7 行和第 9 行所示。如果没有 TAM，DFI 的表现仍然远远超过 MLMS [7] (第 3 行和第 9 行)。这一结果证明了所提出的 DFIM 和 TAM 的有效性。

PR 曲线. 除了数值结果，我们还在 Fig. 7中展示了所提出的方法在这 6 个数据集上的 PR 曲线。可以看出，DFI 的 PR 曲线 (红色实线) 与以前的其他方法相当，在一些数据集上甚至更好。特别是在 PASCAL-S 和 DUTS-TE 数据集上，DFI 比以前的所有方法都要突出。当召回率接近 1 时，我们的准确率远远高于其他方法，这表明我们的显著图中的假阳率很低。

视觉对比. 在 Fig. 6中，我们显示了与以前几种最先进的方法的视觉比较。在最上面一行，显著目标被部分遮挡。DFI 能够分割整个目标，而不会在不相关的区域中混叠。如第二行所示，DFI 还能够以更精确的边界和细节分割出显著目标。当处理显著物体微小且不规则或者前景和背景之间对比度低的图像时，也会发生类似的现象。例如，Fig. 6底部的两行。这些结果表明，DFI 得益于更好地区分边缘像素和分割出整个目标，这可能是联合训练边缘检测和骨架提取任务的优势。

B. 边缘检测

我们将 DFI 与现有的 13 种最先进的检测方法的结果进行比较，包括 gPb-owt-ucm [64], SE-Var [126], MCG [128], DeepEdge [67], DeepContour [66],

表 V
六个广泛使用的数据集上显著目标分割的定量结果。每列的最佳结果以**粗体**突出显示。可以看出，在 F-MEASURE, MAE 和 S-MEASURE 方面我们的方法在几乎所有的数据集上都取得了最好的结果。

	ECSSD [121]			PASCAL-S [122]			DUT-OMRON [123]			HKU-IS [22]			SOD [124]			DUTS-TE [112]		
Model	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$
DCL_{16} [34]	0.896	0.080	0.869	0.805	0.115	0.800	0.733	0.094	0.762	0.893	0.063	0.871	0.831	0.131	0.763	0.786	0.081	0.803
$RFCN_{16}$ [28]	0.898	0.097	0.856	0.827	0.118	0.808	0.747	0.094	0.774	0.895	0.079	0.860	0.805	0.161	0.722	0.786	0.090	0.793
MSR_{17} [35]	0.903	0.059	0.887	0.839	0.083	0.835	0.790	0.073	0.805	0.907	0.043	0.896	0.841	0.111	0.782	0.824	0.062	0.834
DSS_{17} [1]	0.906	0.064	0.880	0.821	0.101	0.804	0.760	0.074	0.789	0.900	0.050	0.881	0.834	0.125	0.764	0.813	0.065	0.826
$NLDF_{17}$ [36]	0.903	0.065	0.870	0.822	0.098	0.805	0.753	0.079	0.770	0.902	0.048	0.878	0.837	0.123	0.759	0.816	0.065	0.816
$Amulet_{17}$ [37]	0.911	0.062	0.876	0.826	0.092	0.816	0.737	0.083	0.784	0.889	0.052	0.866	0.799	0.146	0.729	0.773	0.075	0.800
$PAGR_{18}$ [31]	0.924	0.064	0.883	0.847	0.089	0.822	0.771	0.071	0.775	0.919	0.047	0.889	-	-	-	0.854	0.055	0.839
$DGRL_{18}$ [30]	0.921	0.043	0.899	0.844	0.072	0.836	0.774	0.062	0.806	0.910	0.036	0.895	0.843	0.103	0.774	0.828	0.049	0.842
$MLMS_{19}$ [7]	0.924	0.048	0.905	0.853	0.074	0.844	0.793	0.063	0.809	0.922	0.039	0.907	0.857	0.106	0.790	0.854	0.048	0.862
$JDFPR_{19}$ [33]	0.925	0.052	0.902	0.854	0.082	0.841	0.802	0.057	0.821	-	-	-	0.836	0.121	0.767	0.833	0.058	0.836
$PAGE_{19}$ [50]	0.928	0.046	0.906	0.848	0.076	0.842	0.791	0.062	0.825	0.920	0.036	0.904	0.837	0.110	0.775	0.838	0.051	0.855
$CapSal_{19}$ [53]	-	-	-	0.862	0.073	0.837	-	-	-	0.889	0.058	0.851	-	-	-	0.844	0.060	0.818
CPD_{19} [40]	0.936	0.040	0.913	0.859	0.071	0.848	0.796	0.056	0.825	0.925	0.034	0.907	0.857	0.110	0.771	0.865	0.043	0.869
$PiCANet_{18}$ [45]	0.932	0.048	0.912	0.864	0.075	0.854	0.820	0.064	0.830	0.920	0.044	0.904	0.861	0.103	0.792	0.863	0.050	0.868
$AFNet_{19}$ [51]	0.932	0.045	0.907	0.861	0.070	0.849	0.820	0.057	0.825	0.926	0.036	0.906	-	-	-	0.867	0.045	0.867
$BASNet_{19}$ [54]	0.939	0.040	0.911	0.857	0.076	0.838	0.811	0.057	0.836	0.930	0.033	0.908	0.849	0.112	0.772	0.860	0.047	0.866
DFI (Ours)	**0.945**	**0.038**	**0.921**	**0.880**	**0.065**	**0.865**	**0.829**	**0.055**	**0.839**	**0.934**	**0.031**	**0.919**	**0.878**	**0.100**	**0.802**	**0.888**	**0.038**	**0.887**

表 VI
DFI 与现有边缘检测方法的定量比较。每一列中的最佳结果以**粗体**突出显示。

	BSDS 500 [64]	
Method	ODS $\uparrow$	OIS $\uparrow$
$gPb\text{-}owt\text{-}ucm_{11}$ [64]	0.726	0.757
$SE\text{-}Var_{15}$ [126]	0.746	0.767
MCG_{17} [128]	0.747	0.779
$DeepEdge_{15}$ [67]	0.753	0.772
$DeepContour_{15}$ [66]	0.756	0.773
HED_{15} [10]	0.788	0.808
$CEDN_{16}$ [70]	0.788	0.804
RDS_{16} [71]	0.792	0.810
COB_{17} [11]	0.793	0.820
RCF_{17} [3]	0.811	0.830
$DCNN{+}sPb_{15}$ [69]	0.813	0.831
CED_{17} [72]	0.815	0.833
$LPCB_{18}$ [129]	0.815	0.834
DFI (Ours)	**0.819**	**0.836**

表 VII
DFI 与现有骨骼提取方法的定量比较。每一列中的最佳结果以**粗体**突出显示。

	SK-LARGE [14]	SYM-PASCAL [13]
Method	$F_m\uparrow$	$F_m\uparrow$
MIL_{12} [77]	0.353	0.174
HED_{15} [10]	0.497	0.369
RCF_{17} [3]	0.626	0.392
$FSDS_{16}$ [12]	0.633	0.418
$LMSDS_{17}$ [14]	0.649	-
SRN_{17} [13]	0.678	0.443
LSN_{18} [130]	0.668	0.425
$Hi\text{-}Fi_{18}$ [5]	0.724	0.454
$DeepFlux_{19}$ [81]	0.732	0.502
DFI (Ours)	**0.751**	**0.511**

HED [10], CEDN [70], RDS [71], COB [11], RCF [3], DCNN+sPb [69], CED [72] 和 LPCB [129]，其中大多数是基于 CNN 的方法。

定量分析. 在 Tab. VI，我们展示定量结果。DFI 得到了 0.819 的 ODS、0.836 的 OIS，这甚至比以前那些为边缘检测而精心设计的工作还要好。得益于了 DFIM 和 TAM，来自其他任务的信息不仅不会影响而且还有助于提升边缘检测的性能，如 Tab. III的 “Edge” 列的第 1 行和第 7 行所示。

PR 曲线. 在 BSDS 500 数据集 [64] 上，我们的方法和一些选定方法的精确度-召回率曲线可以在 Fig. 8中找到。可以观察到，由我们的方法产生的 PR 曲线在某

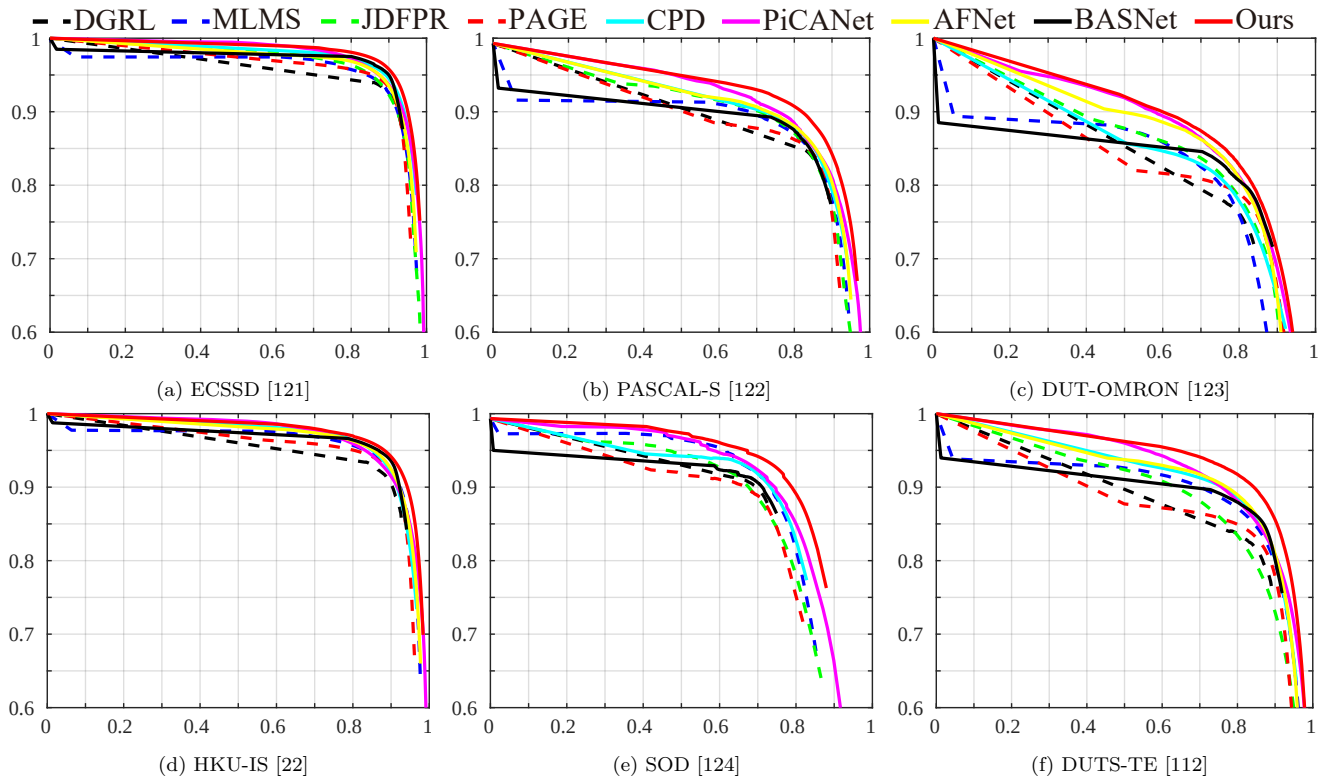


图 7. 六个常用显著目标分割数据集上的精确度 (纵轴) 召回率 (横轴) 曲线。

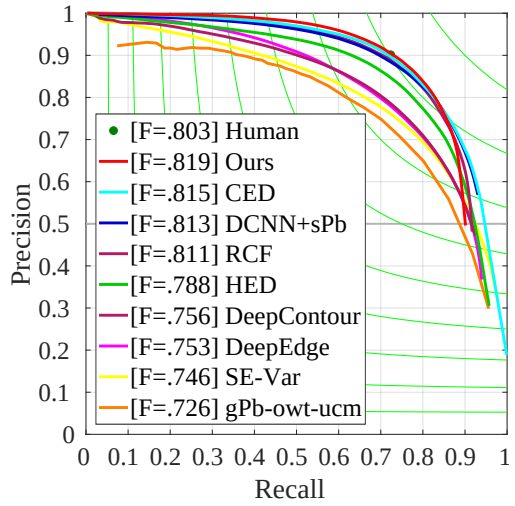


图 8. BSDS 500 数据集的精确度-召回率曲线 [64]。

表 VIII
DFI 和之前最先进方法的平均速度 (FPS) 比较。

	DFI(Multi)	**DFI(Single)**	BASNet [54]	AFNet [51]	PiCANet [45]
Size	400 × 300	400 × 300	256 × 256	224 × 224	224 × 224
Speed	40	57	25	26	7
	PAGE [50]	CPD [40]	DGRL [30]	Amulet [37]	DSS [1]
Size	224 × 224	352 × 352	384 × 384	256 × 256	400 × 300
Speed	25	61	8	16	12
	RCF [3]	CED [72]	LPCB [129]	Hi-Fi [5]	DeepFlux [81]
Size	480 × 320	480 × 320	480 × 320	300 × 200	300 × 200
Speed	36	35	35	32	55

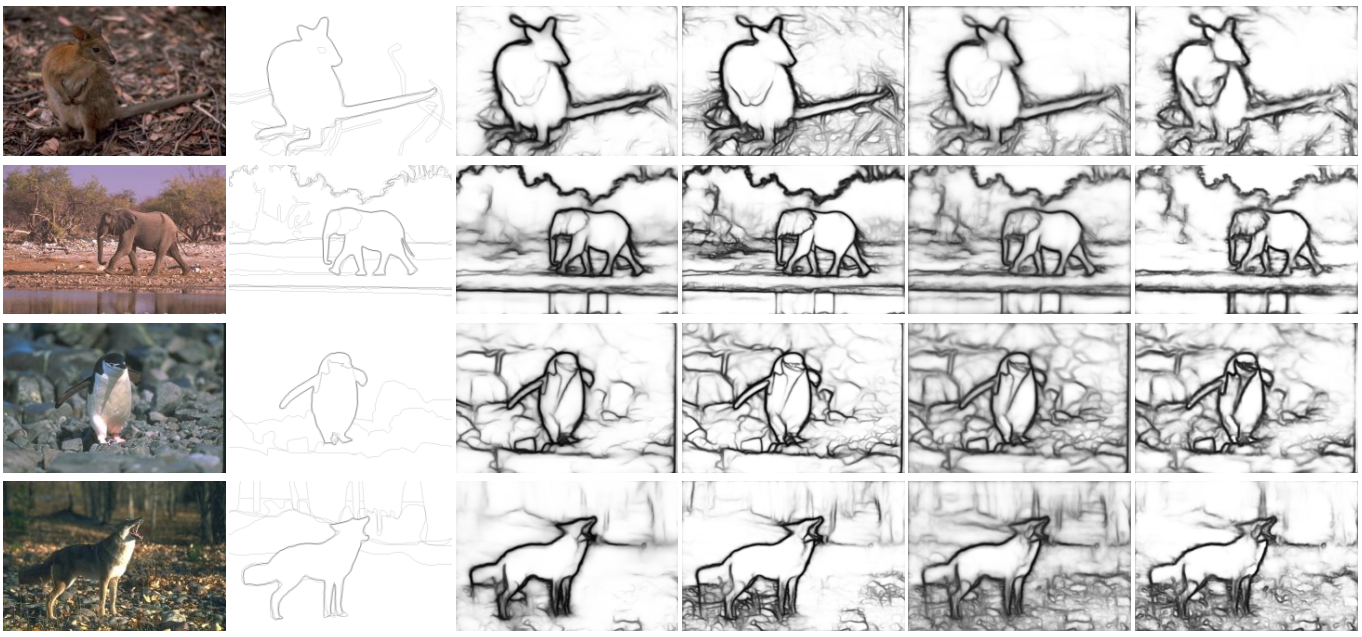
C. 骨架提取

我们比较了 DFI 和最近 9 个基于 CNN 的方法，包括 MIL [77], HED [10], RCF [3], FSDS [12], LMSDS [14], SRN [13], LSN [130], Hi-Fi [5], 和 DeepFlux [81]，比较在 2 个流行的且具有挑战性的数据集上进行，包括 SK-LAGE [14] 和 SYM-PASCAL [13]。为了公平比较，我们分别使用这两个数据集来训练两个不同的模型，如上面的方法中所做的那样。

定量分析. 在 Tab. VII中, 我们展示了与现有方法的定量比较。可以看出，DFI 在 SK-LARGE 数据集上以大比分 (1.9 分) 领先于其余方法 [12]。在 SYM-PASCAL 数据集 [13] 上也有 0.9 分的改进。

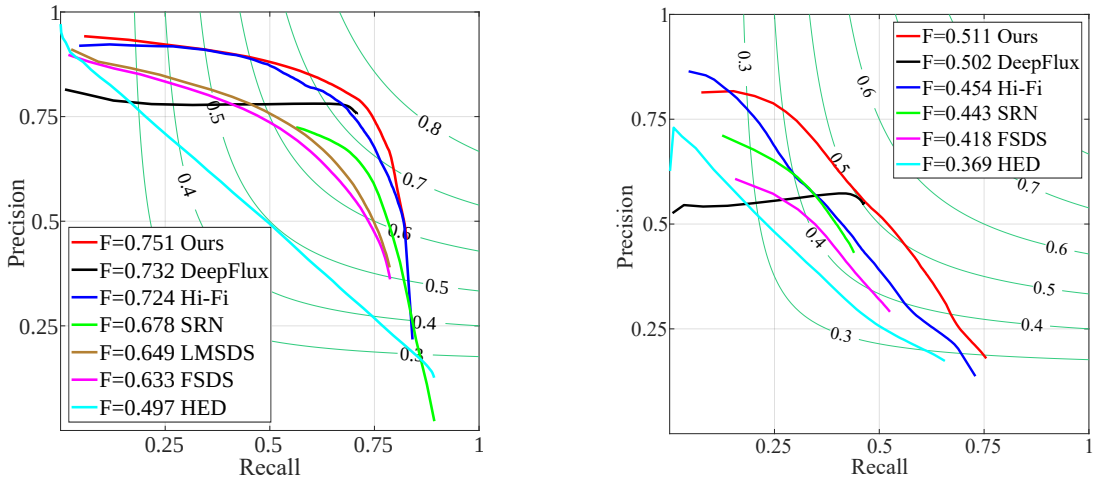
些情况下已经优于人类，并且与以前的方法表现相当，尤其是在精度方面。

视觉分析. 在 Fig. 9中，我们展示了 DFI 和一些主要的代表性方法 [10], [71], [72] 之间的一些直观比较。可以观察到，DFI 在检测边界方面比其他方法表现得更好。在 Fig. 9的最后一行，显然狼的真实边界被很好地突显了。此外，由于动态融合机制，与 [10], [71] 相比 DFI 学习的特征强大得多。这是因为没有边缘的区域描绘得更清晰。总而言之，尽管 ODS 和 OIS 有所改进，但我们的结果在视觉质量上的提升要高得多。



(a) Image (b) GT (c) Ours (d) CED [72] (e) RCF [3] (f) HED [10]

图 9. 与最近几个最先进的边缘检测器进行可视化比较。可以看出，与其他方法相比，DFI 不仅能够生成更清晰的背景，而且在物体边界检测上表现更好。



(a) SK-LARGE [14] (b) SYM-PASCAL [13]

图 10. SK-LARGE 数据集 [14] 和 SYM-PASCAL 数据集 [13] 上一些选定骨架提取方法的精确度-召回率曲线。

PR 曲线. 在 Fig. 10中，我们还展示了我们的方法和一些选定的骨架提取方法的精确度-召回率曲线。可以定量地可以看出，我们的方法在这两个数据集上的表现明显优于其他现有方法。

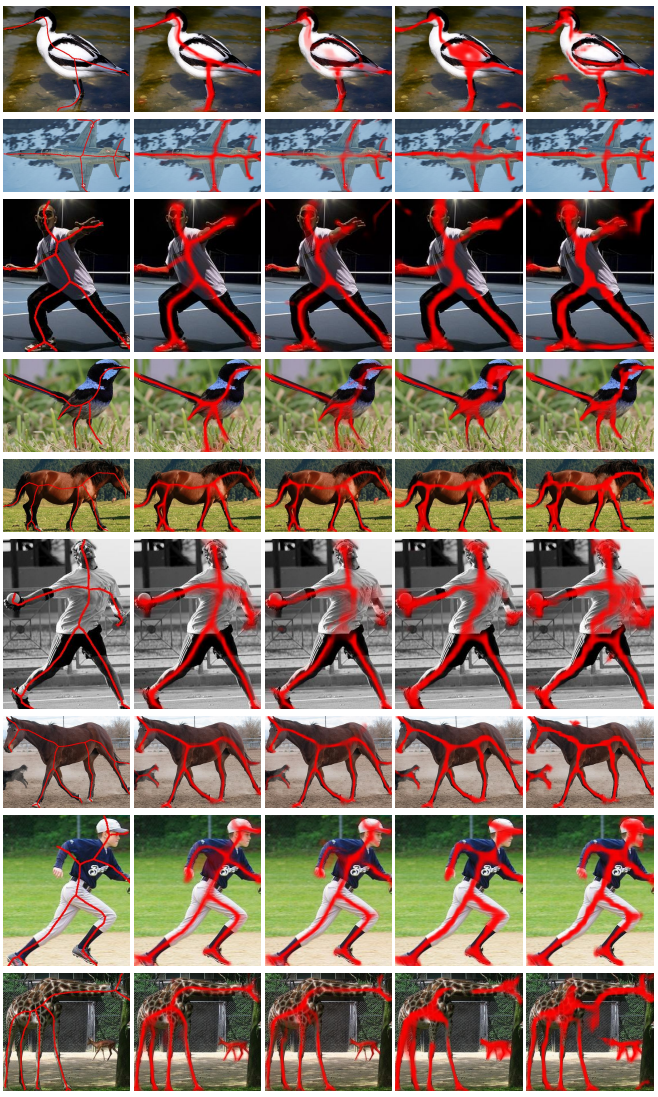
视觉分析. 在 Fig. 11中，我们显示了一些视觉比较。由于动态执行的高级特征集成策略，DFI 能够更精确地定位骨架的准确位置。我们的预测图比其他工作薄得多、强得多的事实也能证实这一点。定量和可视化结果表明，DFI 提供了一种更好的方法来结合不同级别的特征进行骨骼提取，即使是在多任务的方法中。

D. 运行时间的比较

如 Tab. VIII所示，我们将 DFI 的速度与我们论文中评估的其他开源方法进行比较，包括所有三个任务。我们在下面报告了不同方法的平均速度 (fps) 以及相应的输入大小 (在相同的环境中测试)。DFI 可以在单任务模式下以 57 FPS 运行，这与其他方法相当，同时产生更好的检测结果。此外，DFI 即使在多任务模式下也能以 40 帧/秒的速度运行，这意味着同时预测三个不同的任务。

E. 训练时间的消融研究

没有同时执行这三项任务的现有方法。为了突出所提出的 DFIM 和 TAM 带来的影响，我们将该方法



(a) GT (b) Ours (c) Hi-Fi [5] (d) SRN [13] (e) FSDS [12]

图 11. 与三种最近具有代表性的骨架提取方法的可视化比较。很容易发现，我们的结果比其他三种方法要薄得多，也强健得多。此外，由我们的结果产生的骨架是连续的，这对于它们的应用是必不可少的。

表 IX
IMAGENET 预训练影响的消融分析。

Schedule	ImageNet Pre-training	Saliency			Edge		Skeleton
		DUTS-TE			BSDS 500		SK-LAR
		$F_\beta \uparrow$	MAE $\downarrow$	$S_m \uparrow$	ODS $\uparrow$	OIS $\uparrow$	$F_m \uparrow$
1×	w/o	0.819	0.064	0.831	0.786	0.809	0.663
4×	w/o	0.841	0.053	0.848	0.799	0.820	0.738
1×	w/	**0.888**	**0.038**	**0.887**	**0.819**	**0.836**	**0.751**

与其基准版本进行了比较 (Tab. III中第 7 行 *vs.* 第 3 行)，以便证明。所提出的方法需要大约 30 个小时来训练，而基准方法需要 25 个小时。所提出的方法在所有任务中具有更好的、平衡的整体性能，同时 DFIM 和 TAM 的额外参数增加了约 20% 的训练时间。

F. ImageNet 预训练的影响

在上面的部分中，我们使用 ImageNet 预训练作为三个任务的之前的方法来进行实验，以进行公平的比较。在这里，我们研究了使用 ImageNet 预训练对所提出方法的整体性能的影响。从头开始训练时，网络中的所有参数都是随机初始化的。而其他所有的训练设置除了特别声明外的都是一样的。通过比较 Tab. IX的第 1 行和第 3 行，我们可以看到，在进行 1× 周期 (∼12 代) 的训练时，w/o ImageNet 预训练版的整体性能要比 w/ ImageNet 预训练版差很多。即使训练了 4× 周期 (∼ 48 代，36 代后学习率除以 10)，整体表现还是有明显差距的。在训练过程中，我们观察到，当使用 ImageNet 预训练时，损失在早期快速降低和收敛，而随机初始化的版本需要更久的迭代来收敛。ImageNet 数据集有 ∼1.28M(1,281,167) 张图像，比这三个任务所用的图像数量 (21,702, 如 Tab. I中所述) 大 ∼ 59× 倍。从头开始训练的时候，仅仅用 ∼22K 的图像想要很好地优化一个网络是不够的。我们认为，这三个任务都以自然图像作为输入，其中 ImageNet 预训练有助于在训练开始时提供强大的特征提取功能。当从头开始训练时，模型必须学习如何有效地提取特征，并且需要更多的迭代来收敛。通过扩展训练周期，随机初始化的模型最终可能收敛，但特征提取能力的缺乏导致的差距并不易缩小。

VII. 结论

在本文中，我们同时解决了三个不同的低级像素级预测任务，包括显著目标分割、边缘检测和骨架提取。我们提出了一个动态特征整合模块 (DFIM) 来动态地学习每个任务的特征整合策略，以及一个任务自适应注意模块 (TAM) 来跨任务分配信息以获得更好的整体收敛性。在大量数据集上的实验表明，DFI 与已解决任务的最先进方法的性能相当，DFI 的性能有时甚至更好。DFI 的速度也很快，它能以 40 FPS 的速度同时执行这三个像素级的预测任务。

致谢

本研究得到了第 2018AAA0100400 号重大项目“新一代人工智能”、国家自然科学基金委 (61922046)、国家青年人才支持计划、天津市自然科学基金 (18ZXZNGX00110) 的赞助。

参考文献

[1] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 815–828, 2019.

[2] M.-M. Cheng, Q.-B. Hou, S.-H. Zhang, and P. L. Rosin, "Intelligent visual media processing: When graphics meets vision," *Journal of Computer Science and Technology*, vol. 32, no. 1, pp. 110–121, 2017.

[3] Y. Liu, M.-M. Cheng, X. Hu, J.-W. Bian, L. Zhang, X. Bai, and J. Tang, "Richer convolutional features for edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1939 – 1946, 2019.

[4] M.-M. Cheng, X.-C. Liu, J. Wang, S.-P. Lu, Y.-K. Lai, and P. L. Rosin, "Structure-preserving neural style transfer," *IEEE Trans. Image Process.*, vol. 29, pp. 909–920, 2020.

[5] K. Zhao, W. Shen, S. Gao, D. Li, and M.-M. Cheng, "Hi-Fi: Hierarchical feature integration for skeleton detection," in *Int. Joint Conf. Artif. Intell.*, 2018, pp. 1191–1197.

[6] I. Kokkinos, "Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 6129–6138.

[7] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, "A mutual learning method for salient object detection with intertwined multi-supervision," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[8] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[9] Z. Wu, L. Su, and Q. Huang, "Stacked cross refinement network for edge-aware salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[10] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Int. Conf. Comput. Vis.*, 2015, pp. 1395–1403.

[11] K.-K. Maninis, J. Pont-Tuset, P. Arbelaez, and L. Van Gool, "Convolutional oriented boundaries: From image segmentation to high-level tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017.

[12] W. Shen, K. Zhao, Y. Jiang, Y. Wang, Z. Zhang, and X. Bai, "Object skeleton extraction in natural images by fusing scale-associated deep side outputs," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 222–230.

[13] W. Ke, J. Chen, J. Jiao, G. Zhao, and Q. Ye, "Srn: Side-output residual network for object symmetry detection in the wild," *arXiv preprint arXiv:1703.02243*, 2017.

[14] W. Shen, K. Zhao, Y. Jiang, Y. Wang, X. Bai, and A. Yuille, "Deepskeleton: Learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images," *IEEE Trans. Image Process.*, vol. 26, no. 11, pp. 5298–5311, 2017.

[15] M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015.

[16] X. Huang and Y.-J. Zhang, "300-fps salient object detection via minimum directional contrast," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4243–4254, 2017.

[17] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, 2011.

[18] J. Wang, H. Jiang, Z. Yuan, M.-M. Cheng, X. Hu, and N. Zheng, "Salient object detection: A discriminative regional feature integration approach," *International Journal of Computer Vision*, vol. 123, no. 2, pp. 251–268, 2017.

[19] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 733–740.

[20] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Computational Visual Media*, vol. 5, no. 2, pp. 117–150, 2019.

[21] M.-M. Cheng, N. Mitra, X. Huang, and S.-M. Hu, "Salientshape: group saliency in image collections," *The Visual Computer*, vol. 30, no. 4, pp. 443–453, 2014.

[22] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 5455–5463.

[23] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 3183–3192.

[24] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 1265–1274.

[25] R. Fan, M.-M. Cheng, Q. Hou, T.-J. Mu, J. Wang, and S.-M. Hu, "S4net: Single stage salient-instance segmentation," *Computational Visual Media*, vol. 6, no. 2, pp. 191–204, June 2020.

[26] L. Gayoung, T. Yu-Wing, and K. Junmo, "Deep saliency with encoded low level distance map and high level features," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[27] S. He, R. Lau, W. Liu, Z. Huang, and Q. Yang, "Supercnn: A superpixelwise convolutional neural network for salient object detection," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 330–344, 2015.

[28] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Eur. Conf. Comput. Vis.*, 2016.

[29] N. Liu and J. Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[30] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 3127–3135.

[31] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 714–722.

[32] W. Wang, J. Shen, M.-M. Cheng, and L. Shao, "An iterative and cooperative top-down and bottom-up inference network for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[33] Y. Xu, D. Xu, X. Hong, W. Ouyang, R. Ji, M. Xu, and G. Zhao, "Structured modeling of joint deep feature and prediction refinement for salient object detection," in *ICCV*, 2019.

[34] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[35] G. Li, Y. Xie, L. Lin, and Y. Yu, "Instance-level salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[36] Z. Luo, A. K. Mishra, A. Achkar, J. A. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection." in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[37] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Int. Conf. Comput. Vis.*, 2017.

[38] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Int. Conf. Comput. Vis.*, 2017.

[39] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 1741–1750.

[40] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[41] Y. Xu, D. Xu, X. Hong, W. Ouyang, R. Ji, M. Xu, and G. Zhao, "Structured modeling of joint deep feature and prediction refinement for salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[42] Y. Zeng, P. Zhang, J. Zhang, Z. Lin, and H. Lu, "Towards high-resolution salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[43] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[44] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 234–250.

[45] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 3089–3098.

[46] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[47] H. Xiao, J. Feng, Y. Wei, M. Zhang, and S. Yan, "Deep salient object detection with dense connections and distraction diagnosis," *IEEE Transactions on Multimedia*, 2018.

[48] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *Int. Conf. Comput. Vis.*, 2017, pp. 4019–4028.

[49] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 355–370.

[50] W. Wang, S. Zhao, J. Shen, S. C. H. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[51] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[52] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, and A. Borji, "Salient objects in clutter: Bringing salient object detection to the foreground," in *Eur. Conf. Comput. Vis.*, 2018.

[53] L. Zhang, J. Zhang, Z. Lin, H. Lu, and Y. He, "Capsal: Leveraging captioning to boost semantics for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[54] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[55] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[56] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[57] K. Zhao, S. Gao, W. Wang, and M.-M. Cheng, "Optimizing the f-measure for threshold-free salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[58] Y. Liu, Q. Zhang, D. Zhang, and J. Han, "Employing deep part-object relationships for salient object detection," in *Int. Conf. Comput. Vis.*, 2019.

[59] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 6, pp. 679–698, 1986.

[60] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.

[61] V. Torre and T. A. Poggio, "On edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 2, pp. 147–163, 1986.

[62] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu, "Statistical edge detection: Learning and evaluating edge cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 1, pp. 57–74, 2003.

[63] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, 2004.

[64] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2011.

[65] Y. Ganin and V. Lempitsky, "N^ 4-fields: Neural network nearest neighbor fields for image transforms," in *Asian Conf. Comput. Vis.* Springer, 2014, pp. 536–551.

[66] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 3982–3991.

[67] G. Bertasius, J. Shi, and L. Torresani, "Deepedge: A multi-scale bifurcated deep network for top-down contour detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 4380–4389.

[68] J.-J. Hwang and T.-L. Liu, "Pixel-wise deep learning for contour detection," in *ICLR*, 2015.

[69] I. Kokkinos, "Pushing the boundaries of boundary detection using deep learning," *arXiv preprint arXiv:1511.07386*, 2015.

[70] J. Yang, B. Price, S. Cohen, H. Lee, and M.-H. Yang, "Object contour detection with a fully convolutional encoder-decoder network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 193–202.

[71] Y. Liu and M. S. Lew, "Learning relaxed deep supervision for better edge detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 231–240.

[72] Y. Wang, X. Zhao, and K. Huang, "Deep crisp boundaries," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 3892–3900.

[73] J. He, S. Zhang, M. Yang, Y. Shan, and T. Huang, "Bi-directional cascade network for perceptual edge detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[74] Z. Yu and C. Bajaj, "A segmentation-free approach for skeletonization of gray-scale images via anisotropic vector diffusion," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2004.

[75] J.-H. Jang and K.-S. Hong, "A pseudo-distance map for the segmentation-free skeletonization of gray-scale images," in *Int. Conf. Comput. Vis.* IEEE, 2001.

[76] P. Majer, "On the influence of scale selection on feature detection for the case of linelike structures," *Int. J. Comput. Vis.*, vol. 60, no. 3, pp. 191–202, 2004.

[77] S. Tsogkas and I. Kokkinos, "Learning-based symmetry detection in natural images," in *Eur. Conf. Comput. Vis.* Springer, 2012, pp. 41–54.

[78] A. Sironi, V. Lepetit, and P. Fua, "Multiscale centerline detection by learning a scale-space distance transform," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014.

[79] A. Levinshtein, C. Sminchisescu, and S. Dickinson, "Multiscale symmetric part detection and grouping," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 117–134, 2013.

[80] N. Widynski, A. Moevus, and M. Mignotte, "Local symmetry detection in natural images using a particle filtering approach," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5309–5322, 2014.

[81] Y. Wang, Y. Xu, S. Tsogkas, X. Bai, S. Dickinson, and K. Siddiqi, "Deepflux for skeletons in the wild," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 5287–5296.

[82] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[83] C. Doersch and A. Zisserman, "Multi-task self-supervised visual learning," in *Int. Conf. Comput. Vis.*, 2017, pp. 2051–2060.

[84] T. Evgeniou and M. Pontil, "Regularized multi–task learning," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2004, pp. 109–117.

[85] A. Kumar and H. Daume III, "Learning task grouping and overlap in multi-task learning," *arXiv preprint arXiv:1206.6417*, 2012.

[86] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 3994–4003.

[87] C. Ahn, E. Kim, and S. Oh, "Deep elastic networks with model selection for multi-task learning," in *Int. Conf. Comput. Vis.*, 2019.

[88] G. Strezoski, N. v. Noord, and M. Worring, "Many task learning with task routing," in *Int. Conf. Comput. Vis.*, 2019.

[89] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7482–7491.

[90] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Int. Conf. Mach. Learn.*, 2018, pp. 793–802.

[91] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 1871–1880.

[92] S.-A. Rebuffi, H. Bilen, and A. Vedaldi, "Learning multiple visual domains with residual adapters," in *Adv. Neural Inform. Process. Syst.*, 2017, pp. 506–516.

[93] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *Int. Conf. Learn. Represent.*, 2014.

[94] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Adv. Neural Inform. Process. Syst.*, 2015, pp. 91–99.

[95] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[96] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik, "R-cnns for pose estimation and action detection," *arXiv preprint arXiv:1406.5212*, 2014.

[97] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Int. Conf. Comput. Vis.*, 2015, pp. 2938–2946.

[98] K. Du, X. Lin, Y. Sun, and X. Ma, "Crossinfonet: Multi-task information sharing based hand pose estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[99] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Int. Conf. Comput. Vis.*, 2015, pp. 2650–2658.

[100] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun, "Multinet: Real-time joint semantic reasoning for autonomous driving," in *2018 IEEE Intelligent Vehicles Symposium (IV).* IEEE, 2018, pp. 1013–1020.

[101] Y. Gao, J. Ma, M. Zhao, W. Liu, and A. L. Yuille, "Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[102] G.-J. Qi, "Hierarchically gated deep networks for semantic segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 2267–2275.

[103] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, "Gated-scnn: Gated shape cnns for semantic segmentation," in *Int. Conf. Comput. Vis.*, 2019, pp. 5229–5238.

[104] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 2393–2402.

[105] Y. Cheng, R. Cai, Z. Li, X. Zhao, and K. Huang, "Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 3029–3037.

[106] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 840–849.

[107] S. Li, L. Yang, J. Huang, X.-S. Hua, and L. Zhang, "Dynamic anchor feature selection for single-shot object detection," in *Int. Conf. Comput. Vis.*, 2019, pp. 6609–6618.

[108] Z. Chen, Y. Li, S. Bengio, and S. Si, "You look twice: Gaternet for dynamic filter selection in cnns," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 9172–9180.

[109] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 510–519.

[110] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, "Improving convolutional networks with self-calibrated convolutions," in *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2020.

[111] W. Hua, Y. Zhou, C. M. De Sa, Z. Zhang, and G. E. Suh, "Channel gating neural networks," in *Adv. Neural Inform. Process. Syst.*, 2019, pp. 1884–1894.

[112] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 136–145.

[113] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 891–898.

[114] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[115] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[116] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7132–7141.

[117] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[118] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learn. Represent.*, 2015.

[119] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. Neural Inform. Process. Syst.*, 2012.

[120] Y. Wu and K. He, "Group normalization," in *Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[121] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 1155–1162.

[122] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 280–287.

[123] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3166–3173.

[124] V. Movahedi and J. H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 49–56.

[125] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A New Way to Evaluate Foreground Maps," in *Int. Conf. Comput. Vis.*, 2017, pp. 4548–4557.

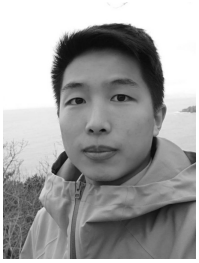
[126] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558–1570, 2015.

[127] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

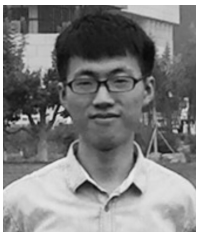
[128] J. Pont-Tuset, P. Arbelaez, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 128–140, 2017.

[129] R. Deng, C. Shen, S. Liu, H. Wang, and X. Liu, "Learning to predict crisp boundaries," in *Eur. Conf. Comput. Vis.*, 2018, pp. 562–578.

[130] C. Liu, W. Ke, F. Qin, and Q. Ye, "Linear span network for object skeleton detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 133–148.



Jiang-Jiang Liu is currently a Ph.D. candidate with School of Computer Science, Nankai University, under the supervision of Prof. Ming-Ming Cheng. His research interests include deep learning, image processing, and computer vision.



Qibin Hou received his PhD degree from Nankai University in 2019. He is currently a research fellow at the Department of Electrical and Computer Engineering, National University of Singapore, working with Prof. Jiashi Feng. His research interests include deep learning and computer vision.

Ming-Ming Cheng received his PhD degree from Tsinghua University in 2012. Then he did 2 years research fellow, with Prof. Philip Torr in Oxford. He is now a professor at Nankai University, leading the Media Computing Lab. His research interests includes computer graphics, computer vision, and image processing. He has published 60+ refereed research papers, with 16, 000+ Google Scholar citations. He received research awards including ACM China Rising Star Award, IBM Global SUR Award, *etc.* He is a senior member of IEEE and on the editor board of IEEE TIP.