基于分层视觉感知学习的轻量级显著性物体检测

刘云 *, 顾宇超 *, 张鑫禹 *, 王薇薇, 程明明

摘要—近来，随着卷积神经网络的迅速发展，显著性物体检测（Salient Object Detection, SOD）取得了巨大进步。然而，SOD 准确率的提升伴随着网络深度和宽度的增加，导致了网络规模和计算开销变大。这使得最新的 SOD 方法无法部署到实际的应用平台上，尤其是移动设备。为了促进 SOD 在现实世界中的应用，本文旨在开发一种轻量级的 SOD 模型。我们发现，灵长类动物视觉系统在不同的视觉皮层区域以不同的感受野和离心率（eccentricities）对视觉信号进行分层处理。受此启发，我们提出了分层视觉感知模块（Hierarchical Visual Perception, HVP）来模仿灵长类动物视觉皮层的分层感知学习。基于 HVP 模块，我们设计了一个轻量级 SOD 网络，命名为 HVPNet。在常用的测试基准上的大量实验表明，所提出的 HVPNet 取得了与最先进的 SOD 方法相当的准确率，并且可以在 CPU 和 GPU 上分别以 4.3fps 和 333.2fps 的速度运行，而参数量只有 1.23M。

***Index Terms*—轻量级显著性物体检测，轻量级显著性检测，分层视觉感知**

I. 引言

人类视觉系统可以在自然图像中迅速、自动地检测到最引人注目的物体或区域。显著性物体检测（Salient Object Detection, SOD）旨在模仿人类的这种本能，以检测图像中最引人注目的区域。SOD 的进步使得许多计算机视觉的应用受益，包括目标检测 [2]、图像检索 [3]、视觉追踪 [4] 和图像缩略图 [5] 等。传统的 SOD 方法 [6], [7] 主要依赖于手工设计的底层特征。尽管高效，由于缺少对高层语义特征的表示能力，这些方法难以建模复杂的自然场景。由于强大的特征学习能力，卷积神经网络（Convolutional Neural Network, CNN），尤其是全卷积神经网络（Fully Convolutional Network, FCN），已经在该领域占据主导地位。大量基于 CNN 和 FCN 的 SOD 方法 [8]–[23] 极大地推动了该领域的发展。

然而，准确率的提升并非没有代价。最近的 SOD 方法需要强大的骨干网络（即编码器）来同时提取底层的、细粒度的细节特征和高层语义特征，还需要一个精心设计的解码器在不损失空间信息的前提下恢复图像分辨率，这两方面均会带来大量的参数和计算开销 [8]–[19], [23]–[33]。相反，近来科研人员对移动应用（如手机）的兴趣日益增加，而受限于计算能力、存储空间和能量供应，移动端设备无法部署以上大规模的 SOD 模型。即便在服务器上部署这些大规模的模型，代价仍然很高。这启发我们在评估 SOD 方法时，速度、网络参数与准确率同样重要。

出于上述考虑，本文旨在设计一种轻量级的 SOD 模型来促进 SOD 的应用。尽管在其他视觉任务中轻量级网络已经被研究过，如图像分类 [34]–[37]，直接将这些骨干网络应用于 SOD 却无法得到满意的结果。这是由于 SOD 对于多尺度学习有特殊的要求（如上所述），而轻量级骨干网络（如 MobileNets [34], [35] 和 ShuffleNets [36], [37]）往往注重于捕获高层语义特征，相对于更深、更宽、有更多卷积核的传统大型网络，其在多尺度学习方面的能力较差。因此，轻量级 SOD 仍是一个具有挑战性的问题，且关键问题在于如何在轻量级的设定下高效地学习多尺度信息。

由于在计算机视觉中模拟人类视觉感知系统用于场景理解是一个大趋势 [38]，我们受灵长类动物视觉系统的启发来解决该问题。在灵长类动物的大脑中，大约 55% 的大脑皮层与视觉有关 [39]，其处理流程为一个分层结构 [40]–[42]。多尺度视觉信号在不同的大脑皮层被层次化地处理，不同的大脑区域有不同的群感受野（population receptive fields, pRF）[43]。Wandell 等人 [43] 发现 pRF 的大小会随着视网膜视位图中的离心率的增加而增大。最近的一项研究 [44] 尝试使用卷积层的卷积核大小和膨胀率分别模拟 pRF 的尺寸和离心率，从而使卷积核大小和膨胀率之间成正相关，该正相关函数类似于 pRF 的尺寸和离心率之间的关系。一个简单的模拟灵长类视觉系统的方法是并行地组

本文是 IEEE T Cybernetics 论文 [1] 的中译版。

织多个 pRF。然而，这忽略了视觉皮层的视觉层次结构，该结构已经在深度学习之前的传统计算机视觉中研究过 [45], [46]。在本文中，我们提出了*分层视觉感知*（Hierarchical Visual Perception, HVP）模块来模拟灵长类动物视觉皮层的结构。HVP 模块使用稠密连接结构和膨胀卷积来分别模拟视觉层次结构和 pRF。实验结果表明，以递减的顺序使用卷积核尺寸和膨胀率会取得最优性能，这与 Hochstein 和 Ahissar 的*反向层次化理论*（Reverse Hierarchy Theory, RHT）[41] 一致，该理论认为视觉感知开始于高层然后传递至低层区域。我们用 HVP 模块和注意力机制设计了轻量级的 SOD 网络，命名为 HVPNet。在常用的基准上的大量实验表明，所提出的 HVPNet 仅用 1.23M 参数取得了与最先进方法相当的精度，而且，以 336×336 尺寸的图像作为输入时，HVPNet 在 CPU 和 GPU 上可以分别达到 4.3fps 和 333.2fps 的速度。

本文的贡献总结如下：

- 为了更好的学习多尺度特征，我们提出了一个*分层视觉感知*（HVP）模块来模拟灵长类动物的视觉层次结构。
- 基于 HVP 模块和注意力机制，我们设计了 HVPNet，据我们所知，这是第一个轻量级的 SOD 卷积神经网络。
- 我们开展了大量实验来探究和评估所提出的 HVPNet，使 HVPNet 可以作为今后轻量级 SOD 研究的一个强大的基准模型。

II. 相关工作

在本节中，我们首先总结 SOD 的最新进展，然后回顾轻量级深度学习的相关文献。

*a) 显著性物体检测：*传统的 SOD 方法 [6], [7], [47]–[49] 主要依赖于手工设计的特征和启发式的先验知识。受限于特征表示能力，手工设计的特征逐渐被深度学习所取代。由于 CNN 和 FCN 的强大的特征学习能力，近五年来，该领域涌现了大量基于 CNN 和 FCN 的方法 [8]–[13], [15], [16], [18]–[33], [50]–[54]。

这些模型大部分主要在研究如何有效地融合多尺度的侧输出信息 [55], [56]。有些方法 [57]–[61] 直接对这些侧输出的特征做级连或加和。有些方法 [62], [63] 使用侧输出特征进行显著性预测，然后融合所有侧输出的预测得到最后的显著性图。大多数方法 [8]–[21], [23]–[32] 使用编码-解码的结构，其中编码器通常是用于图像分类的骨干网络 [64]，解码器负责侧输出特征融合。一些巧妙的设计在该领域吸引了很多注意力。例如，PiCANet [13] 提出使用双向 LSTM 计算全局信息。RAS [62] 提出了一个反向注意力机制来从上到下地融合侧输出特征。

尽管准确率随着网络深度和宽度的增加得到提升，较大的计算开销和网络规模限制了最先进的 SOD 方法在实际系统中的部署，尤其是移动设备。例如，最近的先进方法 EGNet [65] 有 108M 参数，这超过了大部分移动设备可接受的范围。与此方向不同，在本文中，我们为 SOD 开拓了新的方向，即轻量级 SOD，它有巨大的潜力推动 SOD 获得更多实际的应用。我们提出的 HVPNet 取得了与最先进方法相当的精度，同时具有很高的计算效率和很小的网络规模。

*b) 轻量级卷积神经网络：*在许多现实应用中，视觉识别任务必须以实时的、省电的和内存友好的方式在计算资源有限的情况下执行。尽管未被引入 SOD，许多其他视觉任务已经构建了轻量级模型来满足现实部署的要求，方法包括权重量化 [66], [67]、模型压缩 [68], [69] 和高效的网络结构设计 [34]–[37] 等。对于一些视觉任务，如图像分类 [34]–[37]，轻量级网络通过损失一点性能来显著地减小模型大小和浮点运算量 (FLOPs)，从而展现了它们的优越性。MobileNets [34], [35] 利用*深度可分离卷积*和*逐点卷积*的组合来拟合常规卷积的特征表示能力，显著降低了参数量和计算量。基于深度可分离卷积，ShuffleNets [36], [37] 利用随机打乱通道这一操作进一步降低了逐点卷积的参数量和计算量。我们采用与现有技术 [34]–[37] 相同的思想，使用深度可分离卷积来构建我们的模型，而我们主要的技术贡献来自于对层次化的灵长类视觉系统的观察。我们提出了 HVP 模块来模拟灵长类动物视觉的分层结构和 pRF，我们还利用注意力机制来进一步提升性能。基于以上设计，所提出的 HVPNet 在非常轻量化的前提下，取得了与最先进的 SOD 方法相当的性能。

III. 方法

在本节中，我们详述我们的轻量级的 SOD 卷积神经网络结构。具体地说，我们在第III-A节介绍我们从灵长类动物视觉系统中受到的启发。然后，我们在第III-B节中介绍我们主要的模块，*分层视觉感知*（Hierarchical Visual Perception, HVP）模块。其他的网络组成部分以及网络的整体结构分别在第III-C节和第III-D节给出。

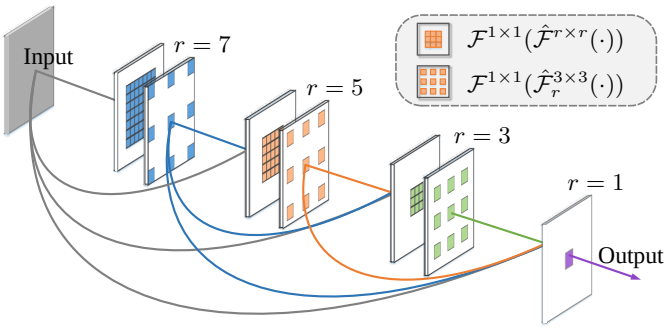


图 1. 所提出的 HVP 模块的图解。

A. 启发和原理

大量神经生理学的证据表明，一系列不同层次的信号处理（8 到 10 层）构成了灵长类动物视觉系统中的分层信号处理系统 [40]–[42]。相比于以并行方式处理信号的所谓扁平处理系统 [46]，分层处理系统展现出明显的优势。事实上，有大量神经生理学的证据表明，认知与深度层次化的概念有关联 [70]。这是符合直觉的，因为我们的眼睛在第一眼看到自然场景时，并不会感知到所有的内容，但会首先识别出与周围环境有较高对比度的物体，这是视觉层次化的简单理解。灵长类视觉系统层次化地处理信息的能力启发了很多计算机视觉方面的研究，请参考 [71] 中的总结。

另外，不同大脑皮层区域的神经元有不同的群感受野（population receptive fields, pRF），而且在每个视网膜视位图中的 pRF 尺寸随着离心率的增大而增大 [43]。视觉系统中 pRF 和离心率的关系可由膨胀卷积 [44] 来模拟。具体来说，我们可以使用卷积核大小和膨胀率分别模拟 pRF 的尺寸和离心率，从而使卷积核大小和膨胀率之间成正相关，该正相关函数类似于 pRF 的尺寸和离心率之间的关系。然而，简单地用扁平处理来学习不同 pRF 的特征（即并联）[46] 并不是最优的，因为它忽略了灵长类动物视觉系统中深度分层这一基本概念。在实验中，我们展示了这种并联设计对于轻量级 SOD 不是最优的。

在本文中，我们提出了一种能更真实地模拟灵长类视觉系统的方法。我们仍然使用膨胀卷积来模拟 pRF。为了模拟 pRF 尺寸和离心率的正相关关系，我们对大的卷积核使用大的膨胀率。对不同的 pRF，我们采用串联而不是简单的并联。由于不知道灵长类动物视觉系统中不同 pRF 之间准确的连接方式，我们提出对不同的 pRF 使用稠密连接，即每个 pRF 的输出特征都将作为后续所有 pRF 的输入。此外，Hochstein 和 Ahissar 提出的反向层次化理论（Reverse Hierarchy Theory, RHT）[41] 称，视觉系统首先在高层产生感知，然后传递至低层，这意味着视觉注意力以从粗粒度到细粒度的方式工作。因此，所提出的 HVP 模块首先使用大的卷积核和膨胀率提取高层信息 (大的 pRF)。直观上，灵长类视觉系统的前注意力部分仅能理解粗略信息。实验结果表明，以递减的方式设置卷积核尺寸和膨胀率具有最好的性能，这验证了我们对 HVP 和 RHT 的假设。因此，所提出的 HVP 模块在理论和实验上都是合理的。

B. 分层视觉感知模块

基于上述原理，我们接下来详细介绍所提出的 HVP 模块。我们采用膨胀卷积来模拟具有不同 pRF 的视觉大脑皮层区域，pRF 的尺寸和离心率与卷积核大小和卷积膨胀率有相似的关系。在此，我们使用深度可分离卷积 (Depth-wise Separable Convolution, DSConv) [34] 和逐点卷积（即 1×1 卷积）作为基本操作来降低参数量和计算量。假设 $\mathcal{F}^{k\times k}$ 代表卷积核大小为 $k\times k$ 的普通卷积。例如，$\mathcal{F}^{1\times1}$ 就是普通的 1×1 卷积。假设 $\hat{\mathcal{F}}_d^{k\times k}$ 代表卷积核大小为 $k\times k$、膨胀率为 d 的 DSConv，$d=1$ 时我们省略下标，即 $\hat{\mathcal{F}}_1^{k\times k}=\hat{\mathcal{F}}^{k\times k}$。

每个 pRF 的模拟单元由一个卷积核大小为 r 的 DSConv 和一个膨胀率大小为 r 的 DSConv 组成，可由如下公式表示

$$\mathcal{R}_r(\boldsymbol{X})=\begin{cases}\mathcal{F}^{1\times1}(\boldsymbol{X}), & if\ r=1;\\ \mathcal{F}^{1\times1}(\hat{\mathcal{F}}_r^{3\times3}(\mathcal{F}^{1\times1}(\hat{\mathcal{F}}^{r\times r}(\boldsymbol{X})))), & if\ r>1,\end{cases} \tag{1}$$

其中，每个卷积层之后都连接了标准的批归一化层 [72] 和 PReLU 层 [73]。此处，我们用 $\hat{\mathcal{F}}^{r\times r}$ 的卷积核大小来模拟 pRF 的尺寸，用 $\hat{\mathcal{F}}_r^{3\times3}$ 的膨胀率来模拟 pRF 的离心率，从而模拟 pRF 尺寸和离心率之间的正相关关系。注意我们使用两个卷积，即 $\hat{\mathcal{F}}^{r\times r}$ 和 $\hat{\mathcal{F}}_r^{3\times3}$，而不是单个卷积 $\hat{\mathcal{F}}_r^{r\times r}$，因为 $\hat{\mathcal{F}}_r^{r\times r}$ 会导致很大的稀疏卷积核（当 $r>3$），不利于网络的训练 [74]，影响推理的效率。使用具有不同 r 的公式 (1)，我们可以模拟灵长类视觉皮层的不同区域，例如 pRF 的大小和离心率都逐渐增加的枕区（occipital areas）V1 至 hV4。从深度学习的角度理解，我们用这种方式可以学习到具有不用感受野的多尺度信息。

如第III-A节所述，不同视觉大脑皮层的处理过程以分层的形式组织。与现有计算机视觉系统 [75], [76] 中所使用的并行处理方式不同，我们使用分层处理的方

式。具体来说，我们以串联方式连接不同 pRF 的模拟单元。此外，视觉大脑皮层中的连接十分复杂，每个区域都与多个其他区域相连。因此，我们对 pRF 模拟单元应用稠密连接 [77] 来模拟视觉皮层的复杂连接，每个 pRF 的输出被作为其他所有后续 pRF 的输入信号。公式上，所有前序 pRF 单元的输出特征被级连起来作为下一单元的输入，即

$$\boldsymbol{X}_i = \mathcal{R}_{r_i}(\text{Concat}(\boldsymbol{X}_0, \boldsymbol{X}_1, \cdots, \boldsymbol{X}_{i-1})), 1 \leq i \leq N, \tag{2}$$

其中 N 为 pRF 单元的个数，$\boldsymbol{X}_0$ (当 $i = 0$ 时的 $\boldsymbol{X}_i$) 代表一个 HVP 模块的输入。Concat$(\cdot)$ 表示级连操作。$\mathcal{R}_{r_i}$ 中第一个 DSConv $\hat{\mathcal{F}}^{r_i \times r_i}(\cdot)$ 的输出通道数和卷积分组数与 $\boldsymbol{X}_0$ 的通道数相等。$\mathcal{R}_{r_i}$ 中其他卷积的通道数与 $\boldsymbol{X}_0$ 相等。从深度学习的角度看，稠密连接可以增加深度并增强特征表示空间，从而带来更好的性能。

最后一个问题是如何确定 pRF 的顺序。我们按照*反向层次化理论* [41] 首先使用大的 pRF 生成视觉感知，然后将感知信号输入小的 pRF。因此，我们以卷积核/膨胀率递减的顺序来组织卷积层，以模拟灵长类动物的视觉系统。例如，对每个 HVP 模块，我们顺序采用 $7, 5, 3, 1$ 的 r 值。直观上，人眼第一眼看到的一般是大物体（大的 pRFs），然后逐渐聚焦到图像细节上。这与用于图像检索和可扩展索引 [78] 的层次化特征表示的原理相似。我们的实验结果也表明，递减顺序的性能比其他顺序要更好。

C. 注意力机制和 *Dropout* 机制

很多工作证明注意力机制在 SOD 中是很有效的 [11], [13], [18], [61], [62], [79]。不同于这些工作仅使用空间注意力机制来自适应地强化或抑制某些空间位置上的信息，我们进一步引入通道注意力来探索通道间的依赖关系并重新校准通道上的激活信息。另外，我们将注意力机制加到编码器网络中，而上述方法只在解码器中使用注意力机制，因为他们的编码器通常使用现有的骨干网络。

a) 通道注意力机制*:* [80] 首次提出了通道注意力机制。设 $\boldsymbol{X} \in \mathbb{R}^{C \times H \times W}$ 为输入激活信号，其中 C、H 和 W 分别为通道数、高度和宽度。我们首先使用全局平均池化（Global Average Pooling, GAP）来提取通道上的特征表示，即

$$\boldsymbol{d}_c = GAP(\boldsymbol{X}) = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{X}_{c,i,j}, \tag{3}$$

其中，$\boldsymbol{d}_c$ 为特征向量 $\boldsymbol{d} \in \mathbb{R}^C$ 的第 c 个值，$\boldsymbol{X}_{c,i,j}$ 是 $\boldsymbol{X}$ 在坐标 (c, i, j) 处的值。然后，我们使用简单的 soft-gating 机制计算每个通道的重要性，即

$$\hat{\boldsymbol{d}} = \sigma(\mathcal{F}^{1\times1}(\psi(\mathcal{F}^{1\times1}(\boldsymbol{d})))), \tag{4}$$

其中内部和外部的 1×1 卷积分别有 $\frac{C}{r}$ 和 C 个输出通道。此处，r 表示减少通道数的比率。因此，我们有 $\hat{\boldsymbol{d}} \in \mathbb{R}^C$。$\psi$ 表示标准的非线性激活函数 [81]，σ 为 *Sigmoid* 函数。随后，将通道注意力和原始特征相乘以校准通道上的激活，即

$$\widetilde{\boldsymbol{X}} = \hat{\boldsymbol{d}} \otimes \boldsymbol{X}, \tag{5}$$

其中，$\hat{\boldsymbol{d}}$ 通过复制变为 $C \times H \times W$ 的大小，$\otimes$ 表示逐元素相乘。

b) 空间注意力机制*:* 给定重新校准后的特征 $\widetilde{\boldsymbol{X}}$，我们基于局部响应提取空间上的重要性。基于轻量级 SOD 的要求，我们的操作是计算高效的。具体来说，我们用具有单个输出通道的 $k \times k$ 卷积来处理 $\widetilde{\boldsymbol{X}}$，并再次使用 soft-gating 机制（即 *Sigmoid*）来计算空间注意力。数学上，该过程可以表示为

$$\boldsymbol{v} = \sigma(\mathcal{F}^{k\times k}(\widetilde{\boldsymbol{X}})), \tag{6}$$

其中，$\boldsymbol{v} \in \mathbb{R}^{H \times W}$。类似地，空间上的重新校准可以表示如下

$$\widehat{\boldsymbol{X}} = \boldsymbol{v} \otimes \widetilde{\boldsymbol{X}}, \tag{7}$$

其中，$\boldsymbol{v}$ 在相乘前通过复制变为 $C \times H \times W$ 的大小。

c) 残差注意力机制*:* 由于我们依次迭代地应用注意力机制，乘以 $(0, 1)$ 范围内的因数会逐渐地弱化激活信息，导致梯度消失问题。为此，我们应用残差学习来促进梯度反传。最后，输出的激活信息变为

$$\boldsymbol{Y} = \widehat{\boldsymbol{X}} + \boldsymbol{X}. \tag{8}$$

d) Dropout: 深度学习中总存在过拟合问题。舍弃部分 CNN 输出激活的策略对于提高泛化能力和避免过拟合被证明是有用的 [82], [83]。在本文中，我们训练时在每个 HVP 模块前连接一个标准的 dropout 比率为 0.1 的 dropout 层。最近基于注意力的 dropout 策略 [83] 根据计算出的显著性图丢弃部分激活信息，不同于此，我们方法中的注意力和 dropout 相互独立以便与先前的 SOD 方法进行公平的比较。换言之，对新的 dropout 策略的探索超出了本文的范围，所以我们按照先前的文献使用标准的 dropout 层 [82]。在测试阶段，dropout 层被直接舍弃。

表 I
所提出的轻量级 SOD 模型的编码器的参数配置。“Module”、“#M”、“#F”、“K” 和 “S” 分别表示模块类型、模块数、输出通道数、卷积核大小和步长。“ResAtt” 代表第III-C节中的残差注意力。

Stage	Resolution	Module	#M	#F	K	S
1	224 × 224	Conv	1	16	3	2
	112 × 112	Conv & ResAtt	1	16	3	1
2	112 × 112	DSConv	1	32	5	2
	56 × 56	HVP & ResAtt	1	32	7-5-3-1	1
3	56 × 56	DSConv	1	64	5	2
	28 × 28	HVP & ResAtt	3	64	7-5-3-1	1
4	28 × 28	DSConv	1	128	5	2
	14 × 14	HVP & ResAtt	5	128	7-5-3-1	1

D. 网络结构

基于以上的模块，我们构建了具有横向连接的编码-解码网络，名为 HVPNet。对于编码器，我们堆叠所提出的 HVP 模块，从而以自底向上的方式快速地提取深度特征。对于解码器，我们以自顶向下的方式融合高层语义特征和低层细节特征。接下来，我们介绍网络设计的细节。

a) 编码器网络: 我们的编码器包含 4 个阶段，每阶段的默认设置总结于表 I。在第 s 阶段，首先使用步长为 2 的卷积（深度可分离卷积或标准卷积）对输入激活信息 $\boldsymbol{F}_{s-1}$ 进行下采样，如下所示

$$\boldsymbol{F}_s = \mathrm{Concat}(\mathcal{H}_s(\boldsymbol{F}_{s-1}), \mathrm{MaxPool}_2(\boldsymbol{F}_{s-1})), \quad (9)$$

其中，$\mathrm{MaxPool}_2$ 代表步长为 2 的最大值池化操作。公式 (9) 之后接有标准的批正则化和 PReLU 层。$\mathcal{H}_s$ 定义为

$$\mathcal{H}_s(\boldsymbol{F}) = \begin{cases} \mathcal{F}^{3\times3}(\boldsymbol{F}), & if\ s = 1; \\ \hat{\mathcal{F}}^{5\times5}(\mathcal{F}^{1\times1}(\boldsymbol{F})), & if\ s > 1. \end{cases} \quad (10)$$

在第一阶段，输入为仅有 3 个通道的彩色图，所以我们直接使用标准卷积。

b) 解码器网络: 给定编码器网络的四个输出特征 $\{\boldsymbol{F}_s : s = 1, 2, 3, 4\}$，解码器的目标是逐步地融合不同尺度的特征，从而逐渐恢复原始分辨率。为了融合顶层特征 $\boldsymbol{F}_3$ 和 $\boldsymbol{F}_4$，我们首先将 $\boldsymbol{F}_4$ 上采样二倍来匹配 $\boldsymbol{F}_3$ 的分辨率，然后使用 1×1 卷积调整特征的通道数。随后，在通道方向将特征图均分为两个特征图，获得的两个特征图在不同尺度上进行微调。可以形式化地表示为

$$\boldsymbol{U}_4 = \mathrm{Upsample}(\boldsymbol{F}_4), \quad (11)$$

$$\boldsymbol{Q}_3 = \mathrm{Concat}(\mathcal{F}^{3\times3}(\widetilde{\boldsymbol{U}}_4^1), \mathcal{F}_2^{3\times3}(\widetilde{\boldsymbol{U}}_4^2)), \quad (12)$$

其中，$\widetilde{\boldsymbol{U}}_4^1$ 和 $\widetilde{\boldsymbol{U}}_4^2$ 代表 $\mathcal{F}^{1\times1}(\boldsymbol{U}_4)$ 的两个分组。最后，使用简单的逐元素相加来融合特征，即

$$\boldsymbol{D}_3 = \psi(\mathrm{BN}(\boldsymbol{Q}_3) + \mathrm{BN}(\mathcal{F}^{1\times1}(\boldsymbol{F}_3))), \quad (13)$$

其中，BN 和 ψ 为批归一化 [72] 和非线性激活函数 [73]。$\boldsymbol{D}_3$ 被传递给下一阶段。对于下面各阶段的特征融合，公式 (11) 被改为 $\boldsymbol{U}_s = \mathrm{Upsample}(\boldsymbol{D}_s)$，其余所有操作均保持不变。最后，我们获得了不同尺度下的融合特征 $\{\boldsymbol{D}_s : s = 1, 2, 3, 4\}$ $(\boldsymbol{D}_4 = \boldsymbol{F}_4)$，这些特征将被用于计算训练的损失。

c) 深监督和损失函数: 我们用深监督 [84] 技术来更好地训练网络。对每个融合特征 $\boldsymbol{D}_s$，$s = 1, 2, 3, 4$，我们使用逐点卷积将其映射成单个通道的特征图，再用 *Sigmoid* 激活函数得到显著性预测 $\boldsymbol{P}_s$。我们使用二元交叉熵（Binary Cross Entropy, BCE）损失进行监督，公式如下

$$\mathcal{L} = \mathrm{BCE}(\boldsymbol{P}_1, \boldsymbol{G}) + \lambda \sum_{s=2}^{4} \mathrm{BCE}(\boldsymbol{P}_s, \boldsymbol{G}), \quad (14)$$

其中，$\boldsymbol{G}$ 代表真值，λ 为超参数，根据经验被设为 0.4 [76]。注意 $\boldsymbol{P}_1$ 为 HVPNet 输出的显著性图。

IV. 实验

A. 实验设置

a) 实现细节: 我们使用 PyTorch 库实现所提出的 HVPNet。默认地，我们使用 Adam 优化器训练模型，权重衰减率设置为 10^{-4}，batch size 为 20。消融实验中，我们的模型及其变体都是从头开始训练 50 个 epoch。与最先进的方法作比较时，我们在 ImageNet 上预训练 HVPNet，就像其他方法的通常做法一样。学习率使用多项式规则进行衰减，即

$$\mathrm{curr\_lr} = \mathrm{init\_lr} \times \left(1 - \frac{\mathrm{curr\_iter}}{\mathrm{max\_iter}}\right)^{\mathrm{power}}, \quad (15)$$

其中，$\mathrm{init\_lr} = 5 \times 10^{-4}$ 且 $\mathrm{power} = 0.9$。

b) 数据集: 我们在六个常用数据集上开展实验，分为是 ECSSD [86]、DUT-O (*i.e.,* DUT-OMRON) [6]、DUTS [87]、HKU-IS [50]、SOD [88] 和 THUR15K [89]。这六个数据集分别包括 1000、5168、15572、4447、300 和 6232 对自然图像和显著性图。DUTS [87] 数据集分为 10553 张训练图像和 5019 张测试图像。根据最近的工作 [8], [13], [58], [60], [65]，我们在 DUTS 训练集上训练模型，并在 DUTS 测试集（DUTS-TE）及其他五个数据集上测试模型。

表 II
与现有 SOD 方法的对比。这里使用 336 × 336 大小的输入计算 FLOPs，除非某方法指定了其输入维度。我们使用黑体标注每列的最佳性能。本文所提出方法的主要优势在于准确率和效率的折中。

Methods	#Param (M)	FLOPs (G)	Speed (FPS)	ECSSD		DUT-O		DUTS-TE		HKU-IS		SOD		THUR15K	
				F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓
DRFI [7]	-	-	0.1	0.777	0.161	0.652	0.138	0.649	0.154	0.774	0.146	0.704	0.217	0.670	0.150
DCL [57]	66.24	224.9	1.4	0.895	0.080	0.733	0.095	0.785	0.082	0.892	0.063	0.831	0.131	0.747	0.096
DHSNet [16]	94.04	15.8	10.0	0.903	0.062	-	-	0.807	0.066	0.889	0.053	0.822	0.128	0.752	0.082
RFCN [24]	134.69	102.8	0.4	0.896	0.097	0.738	0.095	0.782	0.089	0.892	0.080	0.802	0.161	0.754	0.100
NLDF [26]	35.49	263.9	18.5	0.902	0.066	0.753	0.080	0.806	0.065	0.902	0.048	0.837	0.123	0.762	0.080
DSS [63]	62.23	114.6	7.0	0.915	0.056	0.774	0.066	0.827	0.056	0.913	0.041	0.842	0.122	0.770	0.074
Amulet [15]	33.15	45.3	9.7	0.913	0.061	0.743	0.098	0.778	0.085	0.897	0.051	0.795	0.144	0.755	0.094
UCF [25]	23.98	61.4	12.0	0.901	0.071	0.730	0.120	0.772	0.112	0.888	0.062	0.805	0.148	0.758	0.112
SRM [60]	43.74	20.3	12.3	0.914	0.056	0.769	0.069	0.826	0.059	0.906	0.046	0.840	0.126	0.778	0.077
PiCANet [13]	32.85	37.1	5.6	0.923	0.049	0.766	0.068	0.837	0.054	0.916	0.042	0.836	0.102	0.783	0.083
BRN [8]	126.35	24.1	3.6	0.919	0.043	0.774	0.062	0.827	0.050	0.910	0.036	0.843	0.103	0.769	0.076
C2S [10]	137.03	20.5	16.7	0.907	0.057	0.759	0.072	0.811	0.062	0.898	0.046	0.819	0.122	0.775	0.083
RAS [62]	20.13	35.6	20.4	0.916	0.058	0.785	0.063	0.831	0.059	0.913	0.045	0.847	0.123	0.772	0.075
CPD [85]	29.23	59.5	68.0	0.930	0.044	0.794	0.057	0.861	**0.043**	0.924	0.033	0.848	0.113	0.795	0.068
BASNet [29]	87.06	127.3	36.2	**0.938**	**0.040**	**0.805**	**0.056**	0.859	0.048	**0.928**	**0.032**	0.849	0.112	0.783	0.073
EGNet [65]	108.07	270.8	12.7	**0.938**	0.044	0.794	**0.056**	**0.870**	0.044	**0.928**	0.034	**0.859**	**0.110**	**0.800**	**0.070**
HVPNet (OURS)	**1.23**	**1.1**	**333.2**	0.925	0.055	0.799	0.064	0.839	0.058	0.915	0.045	0.826	0.122	0.787	0.076

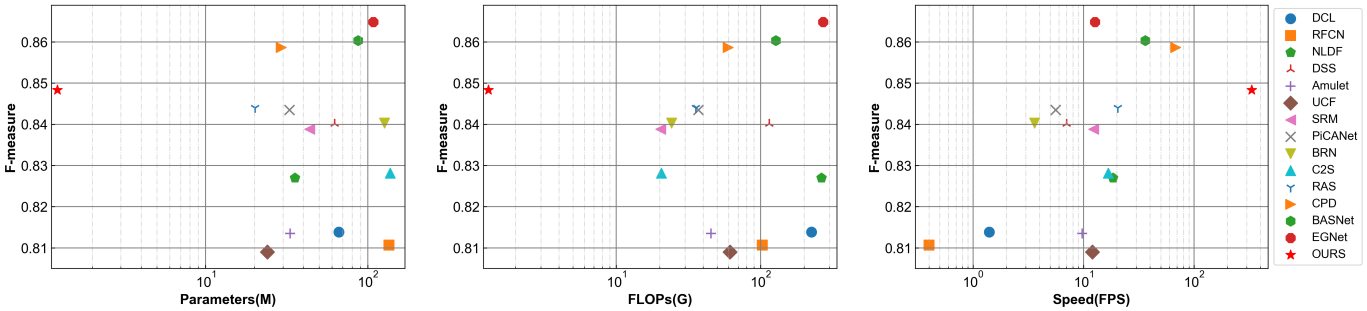


图 2. 表示 F-measure、参数量、FLOPs 和速度的折中的图解。此处，F-measure 是在全部六个数据集上测试结果的均值。注意横坐标值为取对数后的结果。

c) 评估标准: 我们采用两个广泛使用的评估标准，即 F-measure 和平均绝对误差（Mean Absolute Error, MAE）。F-measure（记为 F_β）是基于预测的准确率和召回率来计算的，即

$$F_\beta = \frac{(1+\beta^2) \times \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}}, \tag{16}$$

其中，β^2 通常被设为 0.3，以强调准确率。MAE 为像素级平均绝对误差，计算公式为

$$\text{MAE}(\boldsymbol{P}, \boldsymbol{G}) = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} |\boldsymbol{P}_{ij} - \boldsymbol{G}_{ij}|, \tag{17}$$

其中，$\boldsymbol{P}$ 为预测的显著性图，$\boldsymbol{G}$ 为对应的真值，H 为图像高度，W 为图像宽度。

d) 轻量级的度量标准: 轻量级是本文的关键考虑。这里，我们介绍轻量级的度量标准。如果模型有特定的输入维度，我们就用其默认设定来测试。否则，我们用 336 × 336 的输入大小来测试其速度和计算 FLOPs。本文中的 CPU 速度是在 Intel i7-8700K CPU 上测试的，GPU 速度是在 NVIDIA TITAN Xp GPU 上测试的。

B. 性能比较

a) 与现有 SOD 方法的比较: 我们将所提出的 HVPNet 与 15 个最近的 SOD 方法进行对比。表 II展示了定量的结果。HVPNet 取得了与最先进的 BASNet [29] 和 EGNet [65] 相当的结果，却显著降低了参数量和 Flops。例如，HVPNet 在 ECSSD 上取得了 92.5% 的 F-measure，略低于 EGNet 93.8% 的 F-measure，但 HVPNet 仅需要 EGNet 1.1% 的参数量。HVPNet 还取得了最快的速度和最小的 FLOPs。比如，HVPNet 达到了 333.2fps，而先前方法只能达到 68fps。HVPNet 的 FLOPs 仅为 1.1G。由于 FLOPs 与能量消耗有关，HVPNet 较小的 FLOPs 对移动应用更加友好。

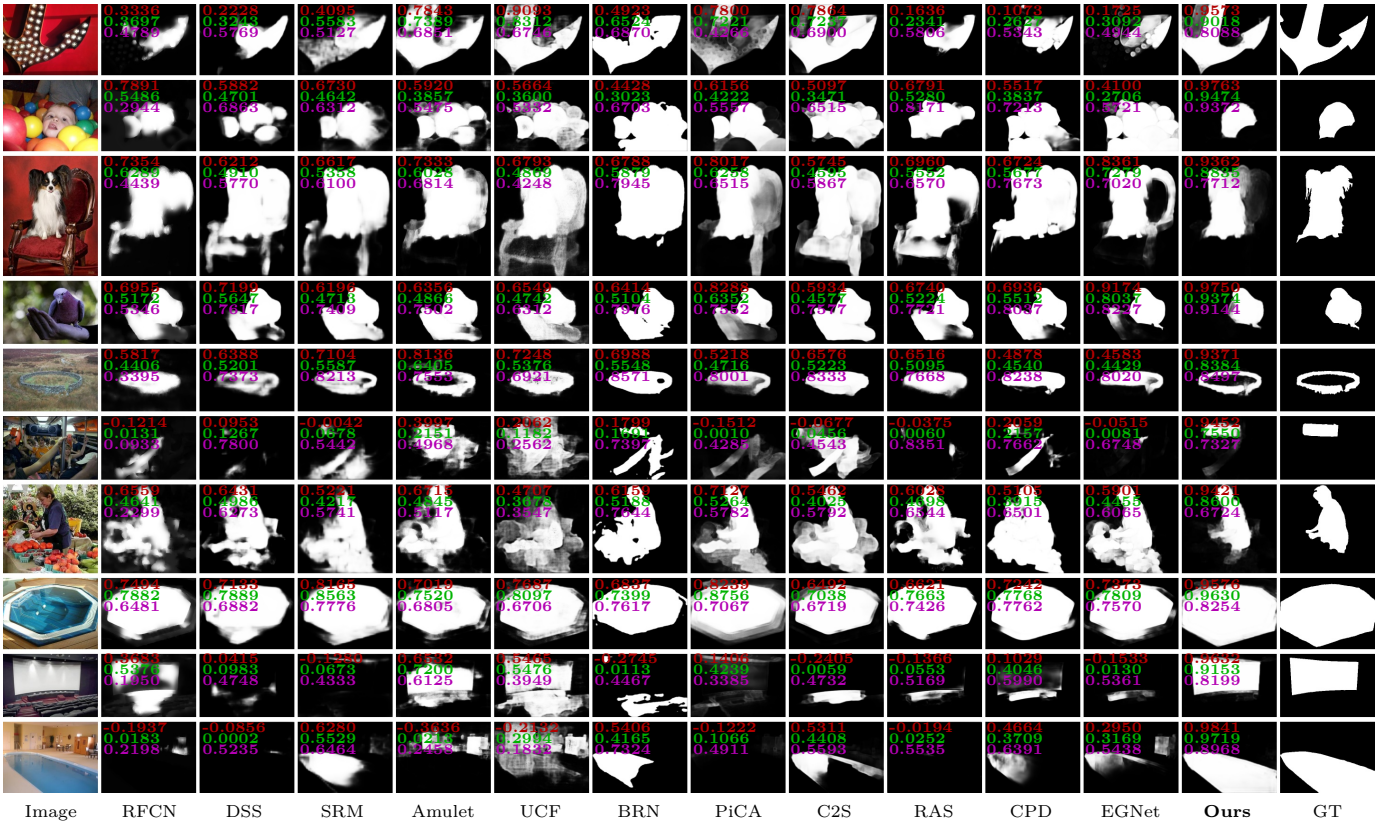


图 3. 与最先进 SOD 方法的定性比较。**红色**、**绿色**和**粉色**数字分别表示预测的显著性图和对应真值之间的 PCC、SIM 和 SSIM 值。

表 III
所提出的 HVPNET 与现有轻量级骨干网络的对比。我们将这些轻量级骨干网络作为编码器，并在其后加上与 HVPNET 相同的解码器，从而用作 SOD。最佳结果用**黑体**标注。

Backbone	ECSSD		DUT-O		DUTS-TE		HKU-IS		SOD		THUR15K	
	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓
MobileNet [34]	0.892	0.127	0.743	0.091	0.792	0.090	0.885	0.098	0.765	0.208	0.759	0.090
MobileNetV2 [35]	0.903	0.066	0.760	0.072	0.804	0.067	0.896	0.053	0.807	0.137	0.768	0.082
ShuffleNet [37]	0.913	0.061	0.764	0.069	0.813	0.063	0.901	0.051	0.815	0.130	0.771	0.080
ShuffleNetV2 [36]	0.898	0.070	0.751	0.076	0.789	0.072	0.885	0.059	0.785	0.147	0.756	0.087
HVPNet (OURS)	**0.925**	**0.055**	**0.799**	**0.064**	**0.839**	**0.058**	**0.915**	**0.045**	**0.826**	**0.122**	**0.787**	**0.076**

为了更好地刻画准确率和效率的折中，我们在图 2中用三个曲线图分别表示 F-measure 与参数量、FLOPs 和速度的关系。为了简洁，我们在所有六个数据集上计算平均 F-measure。在图 F-measure *vs.* parameters 和 F-measure *vs.* FLOPs 中，HVPNet 位于左上方，说明其不仅非常轻量化而且准确率高。在图 F-measure *vs.* speed 中，HVPNet 位于右上角，表明其在准确率和速度之间的良好折中。因此，我们可以得出结论，HVPNet 在准确率、参数量、FLOPs 和速度间取得了良好的折中。

图 3展示了 HVPNet 与其他 SOD 方法的定性比较。为了清晰地展示不同方法预测的显著性图的差异，我们计算了每张预测图与对应的真值显著性图的相似性。我们采用了三种相似性度量指标，包括 Pearson's Correlation Coefficient (PCC 或 CC)、相似性（又称直方图交集，记作 SIM）和 SSIM [90]。请参考关于显著性指标的综述文章 [91] 了解 PCC/CC 和 SIM 的更多细节，而 SSIM [90] 是广为人知的度量结构相似性的指标。从图 3中，我们可以看到，尽管使用极度轻量级设置，HVPNet 仍然在奇异物体（第 1-2 行）、迷惑性场景（第 3-4 行）、细长物体（第 5 行）、复杂背景（第 6-7 行）、不易区分的边界（第 8 行）和大物体（第 9-10 行）等复杂情况下优于先前的方法。这进一步证明了 HVPNet 的优越性。

b) 与轻量级骨干网络的比较: 尽管目前还没有轻量级 SOD 模型，但有用于高效图像分类的轻量级骨

表 IV
消融实验。“PA”、“SE”、“DC”、“CA”、“SA”、“DP” 和 “IP” 分别代表并联、串联、稠密连接、通道注意力、空间注意力、DROPOUT 和 IMAGENET 预训练。

No.	Component							ECSSD		DUT-O		DUTS-TE		HKU-IS		SOD		THUR15K	
	PA	SE	DC	CA	SA	DP	IP	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓
1	✓							0.906	0.065	0.769	0.073	0.799	0.071	0.892	0.056	0.798	0.138	0.759	0.087
2		✓						0.893	0.075	0.750	0.080	0.777	0.080	0.879	0.064	0.792	0.152	0.749	0.091
3		✓	✓					0.909	0.062	0.772	0.072	0.807	0.068	0.899	0.053	0.798	0.138	0.766	0.085
4		✓	✓	✓				0.911	0.062	0.776	0.071	0.807	0.069	0.898	0.053	0.811	0.135	0.765	0.084
5		✓	✓		✓			0.908	0.062	0.768	0.073	0.804	0.069	0.895	0.054	0.809	0.136	0.765	0.085
6		✓	✓	✓	✓			0.912	0.062	0.772	0.071	0.808	0.068	0.898	0.053	0.801	0.138	0.766	0.084
7		✓	✓	✓	✓	✓		0.910	0.065	0.781	0.070	0.814	0.068	0.900	0.054	0.809	0.141	0.769	0.083
8		✓	✓	✓	✓	✓	✓	0.925	0.055	0.799	0.064	0.839	0.058	0.915	0.045	0.826	0.122	0.787	0.076

表 V
消融实验。“KERNEL ORDER” 表示每个 HVP 模块中（公式 (1)）r 的顺序。我们使用 “#MODULES” 表示编码器四个阶段中 HVP 模块的数量。默认设置是按照 $7, 5, 3, 1$ 的顺序设置 r，且模块数量为 $1, 1, 3, 5$。

	Config.	ECSSD		DUT-O		DUTS-TE		HKU-IS		SOD		THUR15K	
		F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓	F_β ↑	MAE ↓
Default Config.		0.910	0.065	0.781	0.070	0.814	0.068	0.900	0.054	0.809	0.141	0.769	0.083
kernel order	$1, 3, 5, 7$	0.901	0.070	0.769	0.078	0.796	0.075	0.891	0.059	0.808	0.138	0.761	0.089
	$1, 5, 3, 7$	0.900	0.070	0.768	0.077	0.790	0.077	0.889	0.060	0.795	0.145	0.762	0.089
	$3, 7, 1, 5$	0.902	0.069	0.765	0.079	0.794	0.076	0.890	0.059	0.795	0.146	0.760	0.089
	$5, 1, 7, 3$	0.905	0.068	0.774	0.076	0.801	0.073	0.892	0.057	0.803	0.137	0.762	0.088
dilation rates	$9, 7, 5, 3, 1$	0.911	0.066	0.778	0.074	0.807	0.071	0.899	0.055	0.814	0.141	0.766	0.087
	$5, 3, 1$	0.910	0.065	0.772	0.075	0.800	0.073	0.894	0.056	0.809	0.141	0.764	0.088
#modules	$1, 1, 2, 2$	0.908	0.065	0.773	0.074	0.800	0.072	0.897	0.055	0.798	0.141	0.765	0.087
	$1, 1, 3, 8$	0.906	0.064	0.772	0.076	0.796	0.073	0.893	0.057	0.801	0.141	0.765	0.089
#filters	$\times 0.75$	0.906	0.067	0.768	0.077	0.796	0.076	0.894	0.058	0.797	0.143	0.763	0.089
	$\times 1.25$	0.915	0.062	0.786	0.072	0.816	0.068	0.902	0.053	0.822	0.137	0.773	0.083

干网络。我们将本文所设计的轻量级解码器与四个轻量级骨干网络组合用于 SOD，包括 MoblieNet [34]、MobileNetV2 [35]、ShuffleNet [37] 和 ShuffleNetV2 [36]。我们采用与 HVPNet 相同的训练设置来训练这些基准网络。表 III展示了评估结果。我们可以发现 HVPNet 比直接在 SOD 中应用轻量级骨干网络取得了更好的结果。这表明轻量级 SOD 是一个值得研究且有前途的研究领域。这同样表明所提出的方法是非平凡的。

C. 消融实验

a) 每个模块的作用: 表 IV验证了 HVPNet 中每个模块的作用。我们从设计并行版本的 HVP 模块开始。我们发现稠密连接的串联版本的 HVP 模块的性能超过并联，这验证了层次化结构的视觉处理感知系统更有效。然后，我们将空间和通道注意力机制整合进我们的编码器中。结果表明，同时加入空间和通道注意力机制对分层视觉感知学习更有益。我们还研究了不同训练策略的影响，即 dropout [82] 和 ImageNet [92] 预训练。我们发现这两种策略都可以在大多数实验设置中提高模型的泛化能力。

b) HVPNet 的设置: 表 V展示了不同网络设置的消融实验的结果。首先，对每个 HVP 模块以递减的方式设置卷积核尺寸的性能明显超过其他版本，表明了*反向层次化理论* [41] 的正确性。其次，增强模型能力，如引入其他 pRF 或增加卷积通道数，可以略微提高性能，但会导致效率下降，这与我们设计的目的相反。考虑到准确率和效率的折中，我们采用表 I中的默认设置。

D. 眼球聚焦预测的评估

另一个与 SOD 高度相关的任务是眼球聚焦预测。不同于 SOD 需要分割出图像中整个的显著性物体，眼球聚焦预测仅需要找到眼球注视点而不需要分割出整个物体。在一些研究中，眼球聚焦预测也被称为显著性预测。在此，我们将其称为眼球聚焦预测，与 SOD 进行区分。为了进一步展示所提出的 HVPNet 的优越性，我们还在著名的 SALICON 2017 [98] 上评估模型对于眼球聚焦的预测能力。SALICON 2017 包括 10000 张训练图像和 5000 张验证图像及其真值标注。测试集包括 5000 张没有真值标签的图像，用于线上竞赛。所有图像的分辨率都是 480×640。我们采用 [95], [96] 中的损失

表 VI
在 SALICON 2017 基准上对眼球聚焦预测的评估。FLOPs 是以 480×640 的图像为输入进行计算的。我们在每一列使用黑体标注最佳结果。所提出的 HVPNet 取得了与最佳性能模型相似的结果，但是具有极度轻量化的设置。

Method	Backbone	Year	#Param (M)	FLOPs (G)	VALIDATION SET				TEST SET			
					NSS ↑	CC ↑	AUC ↑	sAUC ↑	NSS ↑	CC ↑	AUC ↑	sAUC ↑
MLNet [93]	ResNet50	2016	15.42	123.2	1.422	0.584	0.769	0.697	1.453	0.583	0.764	0.687
SalGAN [94]	ResNet50	2017	31.78	94.1	1.635	0.796	0.846	0.716	1.662	0.798	0.847	0.700
SAM [95]	ResNet50	2018	70.09	343.6	1.966	0.900	**0.866**	0.758	1.990	0.899	**0.865**	**0.741**
EML-NET [96]	ResNet50	2018	23.54	25.3	**2.002**	0.879	0.861	0.757	**2.018**	0.874	0.858	0.740
DINet [97]	ResNet50	2019	27.03	156.7	1.957	**0.907**	0.864	**0.759**	1.972	**0.907**	0.863	**0.741**
HVPNet (OURS)	-	-	**1.23**	**3.0**	1.981	0.873	0.865	0.757	2.003	0.869	0.863	0.740

函数，以 batch size 为 8 来训练 HVPNet 10 个 peoch。其他训练设置与 SOD 相同。对于评估指标，我们采用眼球聚焦预测的四个标准的评估指标，包括 NSS、CC、AUC 和 sAUC，使用 SALICON 2017 [98] 提供的代码。关于评估标准的细节，详见综述文章 [91]。

表 VI中展示了评估的结果。我们比较 HVPNet 与最近的眼球聚焦预测方法，包括 MLNet [93]、SalGAN [94]、SAM [95]、EML-NET [96] 和 DINet [97]。可以发现，HVPNet 在所有指标上取得了与最先进方法相似的结果，但采用了极度轻量化设置，即极大地减少了参数量和 FLOPs。因此，我们可以得出结论，HVPNet 在 SOD 和眼球聚焦预测上均取得了效率和性能间的良好折中，这使得其有可能被部署于实际应用中。

V. 结论

本文探索了 SOD 的新方向，即轻量级 SOD，致力于在准确率、效率、参数量和 FLOPs 间取得良好的折中，而不是只关注模型准确率。沿着该思路，我们提出了 HVP 模块来模拟灵长类视觉皮层的分层视觉感知系统。基于 HVP 模块，所提出的 HVPNet 取得了与最先进的 SOD 模型相当的准确率而具有更快的速度、更少的参数量和 FLOPs。据我们所知，这是第一个轻量级 SOD 的尝试。我们还证明在 SOD 中直接应用轻量级骨干网络 [34]–[37] 得不到最优性能，这表明轻量级 SOD 值得作为一个新的研究方向。通过该项研究，我们希望引起大家对轻量级 SOD 更多的关注，从而促进更多实用的 SOD 系统。

致谢

该研究得到了新一代 AI 重大项目（No.2018AAA0100400）、NSFC（61922046）、国家青年人才支持计划和天津自然科学基金（17JCJQJC43700）等的支持。

参考文献

[1] Y. Liu, Y.-C. Gu, X.-Y. Zhang, W. Wang, and M.-M. Cheng, "Lightweight salient object detection via hierarchical visual perception learning," *IEEE Transactions on Cybernetics*, pp. 1–11, 2020.

[2] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 2, 2004, pp. 37–44.

[3] W. Zhou, H. Li, and Q. Tian, "Recent advance in content-based image retrieval: A literature survey," *arXiv preprint arXiv:1706.06064*, 2017.

[4] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognition*, vol. 76, pp. 323–338, 2018.

[5] L. Marchesotti, C. Cifarelli, and G. Csurka, "A framework for visual saliency detection with applications to image thumbnailing," in *Int. Conf. Comput. Vis.*, 2009, pp. 2232–2239.

[6] 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.

[7] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2083–2090.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] W. Wang, J. Shen, X. Dong, and A. Borji, "Salient object detection driven by fixation prediction," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 1711–1720.

[13] 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.

[14] M. A. Islam, M. Kalash, and N. D. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7142–7150.

[15] 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, pp. 202–211.

[16] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 678–686.

[17] S. He, J. Jiao, X. Zhang, G. Han, and R. W. Lau, "Delving into salient object subitizing and detection," in *Int. Conf. Comput. Vis.*, 2017, pp. 1059–1067.

[18] W. Wang, S. Zhao, J. Shen, S. C. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 1448–1457.

[19] 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, pp. 8150–8159.

[20] H. Li, G. Li, B. Yang, G. Chen, L. Lin, and Y. Yu, "Depthwise nonlocal module for fast salient object detection using a single thread," *IEEE Transactions on Cybernetics*, 2020.

[21] S. Chen, B. Wang, X. Tan, and X. Hu, "Embedding attention and residual network for accurate salient object detection," *IEEE Trans. Cybernetics*, vol. 50, no. 5, pp. 2050–2062, 2020.

[22] F. Guo, W. Wang, J. Shen, L. Shao, J. Yang, D. Tao, and Y. Y. Tang, "Video saliency detection using object proposals," *IEEE Transactions on Cybernetics*, vol. 48, no. 11, pp. 3159–3170, 2017.

[23] Y. Zhou, S. Huo, W. Xiang, C. Hou, and S.-Y. Kung, "Semi-supervised salient object detection using a linear feedback control system model," *IEEE Transactions on Cybernetics*, vol. 49, no. 4, pp. 1173–1185, 2018.

[24] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Eur. Conf. Comput. Vis.*, 2016, pp. 825–841.

[25] 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, pp. 212–221.

[26] 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, pp. 6609–6617.

[27] P. Hu, B. Shuai, J. Liu, and G. Wang, "Deep level sets for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 2300–2309.

[28] N. D. Bruce, C. Catton, and S. Janjic, "A deeper look at saliency: Feature contrast, semantics, and beyond," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 516–524.

[29] 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, pp. 7479–7489.

[30] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 1623–1632.

[31] S. Wang, S. Yang, M. Wang, and L. Jiao, "New contour cue-based hybrid sparse learning for salient object detection," *IEEE Transactions on Cybernetics*, 2019.

[32] K. Yan, X. Wang, J. Kim, and D. Feng, "A new aggregation of DNN sparse and dense labeling for saliency detection," *IEEE Transactions on Cybernetics*, 2020.

[33] H. Li, G. Li, and Y. Yu, "ROSA: Robust salient object detection against adversarial attacks," *IEEE Transactions on Cybernetics*, 2019.

[34] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 4510–4520.

[36] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet v2: Practical guidelines for efficient cnn architecture design," in *Eur. Conf. Comput. Vis.*, 2018, pp. 116–131.

[37] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 6848–6856.

[38] I. González-Díaz, V. Buso, and J. Benois-Pineau, "Perceptual modeling in the problem of active object recognition in visual scenes," *Pattern Recogn.*, vol. 56, pp. 129–141, 2016.

[39] D. J. Felleman and D. E. Van, "Distributed hierarchical processing in the primate cerebral cortex," *Cerebral Cortex*, vol. 1, no. 1, pp. 1–47, 1991.

[40] T. Serre, "Hierarchical models of the visual system," *Encyclopedia of Computational Neuroscience*, pp. 1309–1318, 2015.

[41] S. Hochstein and M. Ahissar, "View from the top: Hierarchies and reverse hierarchies in the visual system," *Neuron*, vol. 36, no. 5, pp. 791–804, 2002.

[42] J. J. DiCarlo, D. Zoccolan, and N. C. Rust, "How does the brain solve visual object recognition?" *Neuron*, vol. 73, no. 3, pp. 415–434, 2012.

[43] B. A. Wandell and J. Winawer, "Computational neuroimaging and population receptive fields," *Trends in Cognitive Sciences*, vol. 19, no. 6, pp. 349–357, 2015.

[44] S. Liu, D. Huang, and Y. Wang, "Receptive field block net for accurate and fast object detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 385–400.

[45] L. Wolf, S. Bileschi, and E. Meyers, "Perception strategies in hierarchical vision systems," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 2153–2160.

[46] N. Kruger, P. Janssen, S. Kalkan, M. Lappe, A. Leonardis, J. Piater, A. J. Rodriguez-Sanchez, and L. Wiskott, "Deep hierarchies in the primate visual cortex: What can we learn for computer vision?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1847–1871, 2012.

[47] Q. Wang, Y. Yuan, P. Yan, and X. Li, "Saliency detection by multiple-instance learning," *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 660–672, 2013.

[48] Q. Wang, Y. Yuan, and P. Yan, "Visual saliency by selective contrast," *IEEE Trans. Circ. Syst. Video Technol.*, vol. 23, no. 7, pp. 1150–1155, 2012.

[49] G. Zhu, Q. Wang, Y. Yuan, and P. Yan, "Learning saliency by MRF and differential threshold," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 2032–2043, 2013.

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

[51] Y. Liu, M.-M. Cheng, X. Zhang, G.-Y. Nie, and M. Wang, "DNA: Deeply-supervised nonlinear aggregation for salient object detection," *arXiv preprint arXiv:1903.12476*, 2019.

[52] Y. Qiu, Y. Liu, and J. Xu, "MiniSeg: An extremely minimum network for efficient COVID-19 segmentation," *arXiv preprint arXiv:2004.09750*, 2020.

[53] Y. Qiu, Y. Liu, H. Yang, and J. Xu, "A simple saliency detection approach via automatic top-down feature fusion," *Neurocomputing*, vol. 388, pp. 124–134, 2020.

[54] Y. Qiu, Y. Liu, X. Ma, L. Liu, H. Gao, and J. Xu, "Revisiting multi-level feature fusion: A simple yet effective network for salient object detection," in *IEEE Int. Conf. Image Process.*, 2019, pp. 4010–4014.

[55] 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.

[56] Y. Liu, M.-M. Cheng, D.-P. Fan, L. Zhang, J. Bian, and D. Tao, "Semantic edge detection with diverse deep supervision," *arXiv preprint arXiv:1804.02864*, 2018.

[57] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 478–487.

[58] Y. Zeng, H. Lu, L. Zhang, M. Feng, and A. Borji, "Learning to promote saliency detectors," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 1644–1653.

[59] X. Chen, A. Zheng, J. Li, and F. Lu, "Look, perceive and segment: Finding the salient objects in images via two-stream fixation-semantic CNNs," in *Int. Conf. Comput. Vis.*, 2017, pp. 1050–1058.

[60] 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.

[61] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 3085–3094.

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

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

[64] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.

[65] J.-X. Zhao, 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, pp. 8779–8788.

[66] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 4820–4828.

[67] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-Net: Imagenet classification using binary convolutional neural networks," in *Eur. Conf. Comput. Vis.*, 2016, pp. 525–542.

[68] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in *Int. Conf. Comput. Vis.*, 2017, pp. 2736–2744.

[69] P. Molchanov, A. Mallya, S. Tyree, I. Frosio, and J. Kautz, "Importance estimation for neural network pruning," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 11 264–11 272.

[70] J. B. Tenenbaum, C. Kemp, T. L. Griffiths, and N. D. Goodman, "How to grow a mind: Statistics, structure, and abstraction," *Science*, vol. 331, no. 6022, pp. 1279–1285, 2011.

[71] J. Benois-Pineau and P. Le Callet, *Visual Content Indexing and Retrieval with Psycho-Visual Models*. Springer, 2017.

[72] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[73] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[74] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[75] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2018.

[76] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 2881–2890.

[77] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 4700–4708.

[78] C. Morand, J. Benois-Pineau, J.-P. Domenger, J. Zepeda, E. Kijak, and C. Guillemot, "Scalable object-based video retrieval in HD video databases," *Signal Processing: Image Communication*, vol. 25, no. 6, pp. 450–465, 2010.

[79] Y. Yuan, Z. Xiong, and Q. Wang, "VSSA-NET: Vertical spatial sequence attention network for traffic sign detection," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3423–3434, 2019.

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

[81] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[82] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[83] A. M. Obeso, J. Benois-Pineau, M. S. G. Vázquez, and A. A. R. Acosta, "Dropping activations in convolutional neural networks with visual attention maps," in *Int. Conf. Content-Based Multimedia Indexing*. IEEE, 2019, pp. 1–4.

[84] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *AISTATS*, 2015, pp. 562–570.

[85] 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, pp. 3907–3916.

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

[87] 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.

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

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

[90] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[91] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, "What do different evaluation metrics tell us about saliency models?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 740–757, 2018.

[92] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 248–255.

[93] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "A deep multi-level network for saliency prediction," in *Int. Conf. Pattern Recog.* IEEE, 2016, pp. 3488–3493.

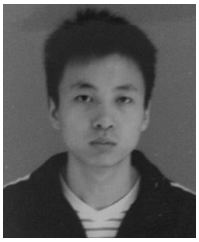
[94] J. Pan, E. Sayrol, X. G.-i. Nieto, C. C. Ferrer, J. Torres, K. McGuinness, and N. E. OConnor, "SalGAN: Visual saliency prediction with adversarial networks," in *CVPR Scene Understanding Workshop (SUNw)*, 2017.

[95] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "Predicting human eye fixations via an LSTM-based saliency attentive model," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5142–5154, 2018.

[96] S. Jia and N. D. Bruce, "EML-NET: An expandable multi-layer network for saliency prediction," *Image and Vision Computing*, p. 103887, 2020.

[97] S. Yang, G. Lin, Q. Jiang, and W. Lin, "A dilated inception network for visual saliency prediction," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 2163–2176, 2019.

[98] M. Jiang, S. Huang, J. Duan, and Q. Zhao, "SALICON: Saliency in context," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 1072–1080.



刘云 是一名南开大学计算机学院的博士生。他在 2016 年于南开大学获得学士学位。他的研究方向包括计算机视觉和机器学习。

顾宇超 是一名南开大学计算机学院的硕士研究生。他在 2019 年于北京化工大学获得学士学位。他的研究方向包括视频分析和迁移学习。

张鑫宇 是一名南开大学数学科学学院的本科生。他的研究方向包括计算机视觉和深度学习。

王薇薇 是一名南开大学数学科学学院的本科生。她的研究方向包括计算机视觉和机器学习。

程明明 在 2012 年于清华大学获得博士学位。随后，他在牛津大学与 Philip Torr 教授合作，作为研究员工作了两年。他现在是南开大学教授，领导媒体计算实验室。他的研究方向包括计算机图形学、计算机视觉和图像处理。他获得的学术荣誉包括 ACM 中国新星奖、IBM 全球 SUR 奖和 CCF-Intel 青年教师研究计划。他是 IEEE TIP 的编辑委员会成员。