

# FlatNet: 通过无镜头测量实现真实感场景重建

Salman Siddique Khan<sup>ID</sup>, Varun Sundar<sup>ID</sup>, Vivek Boominathan<sup>ID</sup>,  
Ashok Veeraraghavan<sup>ID</sup>, and Kaushik Mitra<sup>ID</sup>

**摘要:** 无透镜成像已成为通过避开传统相机中的大体积透镜来实现超小型相机的潜在解决方案。由于没有聚焦镜头, 无镜头相机依靠计算算法从多路测量中恢复场景。然而, 当前基于迭代优化的重建算法会产生噪声更大、感知效果更差的图像。在这项工作中, 我们提出了一种基于非迭代深度学习的重建方法, 该方法可将无透镜重建的图像质量提高几个数量级。我们的方法称为 FlatNet, 它建立了一个框架, 用于从基于掩模的无镜头相机重建高质量的真实感图像, 其中相机的前向模型公式是已知的。FlatNet 包含两个阶段: (1) 反转阶段, 通过学习正向模型公式中的参数将测量映射到中间重建空间; (2) 感知增强阶段, 提高中间重建的感知质量。这些阶段以端到端的方式一起训练。我们通过使用两种不同类型的无透镜原型在真实和具有挑战性的场景上进行广泛的实验来展示高质量的重建: 一种使用可分离的前向模型, 另一种使用更通用的不可分离的裁剪卷积模型。我们的端到端方法速度快, 可以产生逼真的重建, 并且很容易用于其他基于掩模的无镜头相机。

索引术语—无透镜成像、图像重建

## 1 简介

EMERGING 应用, 例如可穿戴设备、增强现实虚拟现实、生物识别和许多其他技术正在推动对高度小型化成像系统的迫切需求。不幸的是, 当前一代相机基于镜头, 而这些镜头通常占相机成本、体积和重量的 90% 以上。虽然镜头和光学器件已经小型化了两个数量级, 但在上个世纪, 我们正在慢慢违反阻碍进一步小型化的基本定律 (衍射极限和洛曼缩放定律 [3])。在过去十年中, 无透镜成像系统已成为轻质、超紧凑、廉价成像的潜在解决方案。无透镜成像的基本思想是用幅度掩模[1]或相位掩模[2]、[4]代替透镜; 通常放置在非常靠近传感器的位置。与基于镜头的相机相比, 这些无镜头成像系统具有许多优势。镜头的需求是决定相机尺寸和重量的主要因素, 因此不再需要镜头。此外, 无透镜设计允许更广泛的传感器几何形状, 允许传感器具有更多非常规形状 (例如球形或圆柱形) 或物理上

灵活[5]。此外, 无镜头相机可以使用传统的半导体制造技术来生产, 因此可以利用其所有的缩放优势 - 生产低成本、高性能的相机 [6]。由于没有任何聚焦元件, 传感器测量记录在无镜头相机中成像仪不再是场景的照片, 而是高度复合的测量结果。需要重建算法来消除这种多路复用的影响并生成正在成像的场景的照片。然而, 无镜头相机的恢复算法的设计是一项具有挑战性的任务, 主要是因为无镜头设计固有的点扩散函数 (PSF) 的大量支持。特别是, 恢复算法面临以下挑战。首先, PSF 的大量支持导致大型线性系统, 这使得此类系统难以存储和反演。其次, 大的 PSF 也会导致非常高程度的全局复用。传统的数据驱动方法, 如专为自然图像设计的卷积神经网络, 由于其有限的感受野, 不适合处理如此大量的复用。第三, 无透镜设计会导致病态系统, 从而影响重建质量以及噪声特性。此类系统的特点。在图 1 所示的吉洪诺夫正则化重建中可以观察到较差的重建质量。因此, 无镜头相机需要强大而高效的算法来克服这些挑战。考虑到上述挑战, 我们提出了一种用于照片级真实感无镜头重建的前馈深度神经网络, 该网络我们称之为 FlatNet。FlatNet 学习从无透镜测量到场景输出的直接映射。FlatNet 包含两个阶段: 第一阶段是可学习的反转阶段, 它将多路复用测量带回图像空间。此阶段取决于相机型号。

- Salman Siddique Khan, Varun Sundar, and Kaushik Mitra are with the Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai, Tamil Nadu 600036, India.  
E-mail: {sk39, varunsundar}@smail.iitm.ac.in, kmitra@ee.iitm.ac.in.
- Vivek Boominathan and Ashok Veeraraghavan are with Rice University, Houston, TX 77005 USA. E-mail: {vivekb, vashok}@rice.edu.

手稿于 2020 年 5 月 28 日收到; 2020 年 9 月 9 日修订; 接受日期: 2020 年 10 月 16 日。出版日期: 2020 年 10 月 26 日; 当前版本的日期为 2022 年 3 月 4 日。(通讯作者: Salman Siddique Khan。) 建议由 K. Nishino 接受。数字对象标识符编号。10.1109/TPAMI.2020.3033882

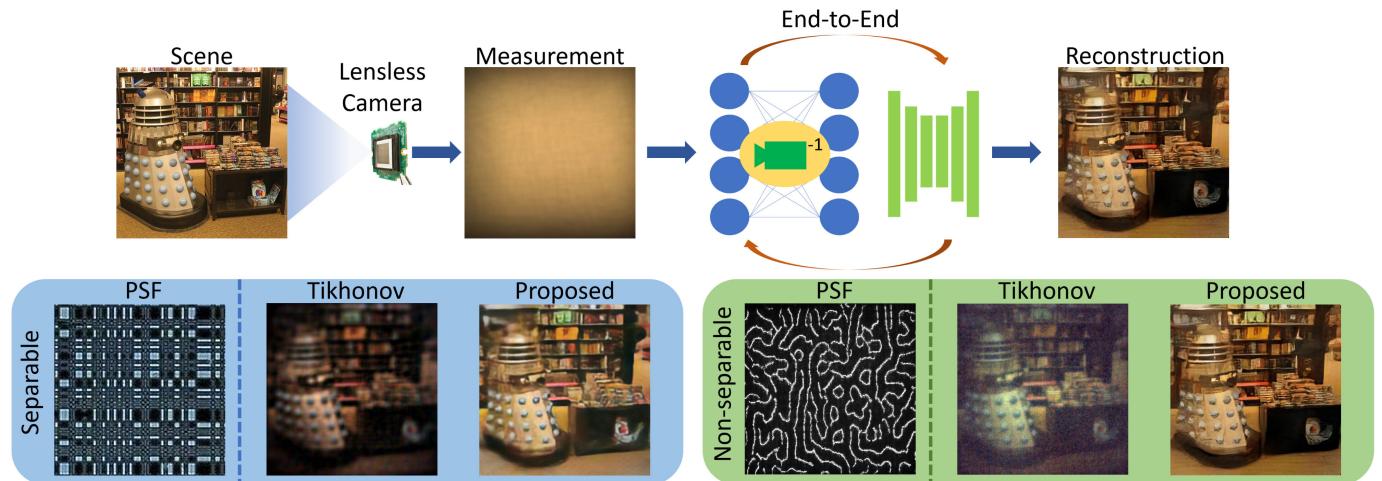


图 1. 无透镜成像。无镜头相机需要计算才能从测量中恢复真实场景。在这项工作中，我们提出了一种基于深度学习的无透镜重建算法，适用于基于可分离 [1] 和不可分离掩模 [2] 的无透镜相机，可为真实且具有挑战性的场景生成逼真的重建。

第二阶段使用全卷积网络增强了中间重建。应该注意的是，这两个阶段都是以端到端的方式进行训练的。文献[2]表明，与现有的不可分离无透镜相机相比，基于可分离无透镜掩模的无透镜相机具有较差的特性。在我们之前的工作 [7] 中，我们已经证明了 FlatNet 对于可分离无透镜模型的有效性。但它不能轻易地用于基于不可分离掩模的无镜头相机。在这里，我们将之前的工作扩展到处理不可分离的无透镜模型。特别是，我们提出了一种基于傅里叶域运算的不可分离无透镜模型的可学习中间映射的有效实现。我们还为这个可学习的中间阶段提出了一种初始化方案，不需要显式的 PSF 校准。我们证明，对于无透镜模型是非循环的情况，中间映射是稳健的。当传感器尺寸小于解卷积所需的完整测量尺寸时，就会发生这种情况。最后，为了验证 FlatNet 的鲁棒性和效率，我们对使用基于可分离掩模的无透镜相机（称为 FlatCam [1]）和基于不可分离掩模的无透镜相机（称为 PhlatCam [2]）捕获的具有挑战性的现实场景进行了广泛的实验。总而言之，本文的主要贡献是：

我们为不可分离或通用无透镜模型的可学习中间阶段提出了一种有效的实现。在[7]中，我们仅针对可分离无镜头模型展示了这一点。在这里，我们将其扩展到一般的无镜头情况。我们验证了所提出的不可分离无透镜模型的可学习中间映射在无透镜系统不遵循完全卷积或循环假设的挑战性场景中的鲁棒性。我们提出了一种不需要显式 PSF 校准的不可分离无透镜模型的初始化方案。与我们之前的工作 [7] 中描述的使用可分离掩模 FlatCam and 收集的显示和直接捕获的测量结果类似，我们收集

不可分离 maskPhlatCam [2] 的相应数据集。我们还收集了一个无约束室内无透镜测量数据集，并与相应的未对齐网络摄像头图像配对，最终用于微调我们提出的 FlatNet，以稳健地处理无约束的现实世界场景。我们的方法优于以前的传统和基于深度学习的无透镜重建方法。

### 1.1 相关工作 1.1.1 无透镜成像

无镜头成像涉及捕获场景的图像，而无需使用镜头对入射光进行物理聚焦。它过去已广泛用于天文学的 X 射线和伽马射线成像 [8]、[9]，但其在可见光谱应用中的使用最近才被研究。在无透镜成像系统中，场景可以直接在传感器上捕获 [10]，也可以在通过掩模元件调制后捕获。已使用的掩模类型包括相位光栅 [11]、随机漫射器 [4]、设计相位掩模 [2]、振幅掩模 [1]、[12]、压缩采样器 [13]、[14] 和空间光调制器 [15]、[16]。用上述掩模替换镜头会导致多路传感器捕获与成像场景缺乏任何相似之处。然后使用计算重建算法恢复可识别图像。在本文中，我们为基于可分离和不可分离掩模的无透镜相机开发了一种基于深度学习的重建算法。

### 1.1.2 图像重建

图像重建是大多数计算成像问题的核心方面 [1]、[2]、[4]、[17]、[18]。一般来说，计算成像的图像重建是不稳定的并且需要正则化。图像重建的传统方法涉及解决正则化最小二乘问题。过去已经开发了许多基于启发式的正则化器。这些包括稀疏梯度域 [2]、[4]、[19]、小波/频域稀疏性 [20] 等。然而，这些方法存在以下问题：

事实上, 生成的成本函数通常不具有封闭形式的最小值, 必须采用迭代方法来解决它。此外, 正则化器基于启发式方法, 对于当前的特定任务可能并不理想。深度神经网络也被设计用于解决计算成像系统中的图像重建问题。一类基于深度学习的解决方案涉及正则化器或近端映射阶段的学习, 然后迭代地解决 MAP 问题。[21]、[22]、[23] 等方法属于此类。另一类算法被设计为前馈深度神经网络, 它以监督或自监督的方式进行训练。压缩图像恢复[24]、[25]、[26]、傅里叶叠层摄影[27]、无透镜恢复[28]都属于这一类别。在这些前馈网络中, [26]、[28]受到成像模型物理的启发, 是传统优化框架的展开版本。尽管这些方法提供了可解释性, 但它们的缺点包括由于计算量增加和内存消耗增加。大量展开的迭代。所提出的方法及其初步版本[7]也属于受物理学启发的深度神经网络的范畴。然而, 它们不涉及任何展开, 从而避免大量的计算和内存成本。

## 2 基于掩模的无透镜成像

基于掩模的无透镜成像仪与基于透镜的成像仪不同, 可测量场景的全局线性复用版本。这种多路复用是放置在传感器前面的掩模的函数。在数学上, 这给出为

$$y = \Phi x + n, \quad (1)$$

其中  $x$  和  $y$  分别是场景和测量的矢量化表示,  $\Phi$  表示广义线性变换,  $n$  是加性噪声。一般来说,  $\Phi$  具有很大的内存占用, 因此, 用  $\Phi$  进行存储和计算在计算上是很困难的。根据传感器测量的  $O \times N \times 2$  像素重建具有  $O \times N \times 2$  像素的场景需要具有  $O \times N \times 4$  像素的  $\Phi$ 。例如, 1 兆像素的场景和 1 兆像素的传感器需要 10 美元的 12 个元件。然而, 通过仔细设计掩码并使用源自物理的前向模型, 可以大大降低计算复杂度。掩码执行的调制表征了线性矩阵  $\Phi$ 。通过使用低秩可分离掩码模式, 可以将巨大的  $\Phi$  表示为分解成更小的矩阵[1], [29]。具体来说, 在[1]中, 单可分离无透镜前向模型简化为

$$Y = \Phi_L X \Phi_R^T + N, \quad (2)$$

其中,  $\Phi_L$  和  $\Phi_R$  是  $\Phi$  的可分离细分,  $X$  是 2D 场景辐照度,  $Y$  是 2D 记录的测量值,  $N$  模型加性噪声。通过在不可分离掩模上添加足够小的孔径, 从而确保离轴移动的 PSF 留在传感器内, [2] 表明无透镜前向模型可以写成卷积模型

$$Y = P * X + N, \quad (3)$$

其中  $P$  是系统的 PSF。无镜头相机的 PSF 是单点光源照射时掩模在传感器上投射的图案[2]。当点源横向移动时, PSF 会发生变化, 对于一般场景, 传感器测量结果是各种移动 PSF 的加权和, 从而形成卷积模型。如果传感器与 PSF 相比不够大, 则 PSF 可能会移出用于倾斜角度场景点的传感器。在这种情况下, [4] 使用裁剪卷积模型

$$Y = C(P * X) + N, \quad (4)$$

其中  $C$  是传感器裁剪操作。等式(4)描述的这种系统不再是循环的。对于可分离掩模, 裁剪已经合并到模型矩阵  $F_L$  和  $F_R$  中。在这项工作中, 我们将主要关注两种无镜头相机原型, (a) FlatCam [1], 它具有可分离掩模 mask 和 (b) PhlatCam [2] 具有不可分离的 mask。我们探索了一种数据驱动的方法, 该方法结合了无透镜成像模型, 从上述相机中生成逼真的重建结果。我们还通过预处理传感器测量来探索 PhlatCam 传感器裁剪的另一种方法[30]。

## 3 FLATNET

为了解决无镜头图像重建中涉及的挑战, 我们采用数据驱动的方法进行场景恢复。我们将重建框架建模为一个两阶段完全可训练的深度网络。然后, 这个两阶段网络在对抗性设置中进行联合训练。该网络如图 2.Trainable Camera Inversion 所示。FlatNet 的第一阶段是可学习的中间映射, 称为可训练相机反转阶段, 它学习反转无透镜前向模型, 从全局多重无透镜测量中获得中间重建。我们利用每种类型的无透镜系统的正向模型的特性, 为可分离和不可分离的无透镜模型实现了可训练反演阶段的单独公式。感知增强。FlatNet 的第二阶段称为感知增强阶段, 是一个全卷积网络, 可增强从可训练反转阶段获得的中间重建, 使其具有更加真实的外观。

### 3.1 可训练的相机反演

在我们网络的第一阶段, 我们学习反转无镜头相机模型的前向操作。这使我们能够获得具有完整局部结构的中间表示。为了实现这一点, 我们对可分离和不可分离无镜头相机模型采用单独的方法。由于可分离模型的计算简单性, 我们将首先描述可分离模型的反演阶段的实现。

#### 3.1.1 可分离模型

给定等式(2)中描述的无透镜模型, 我们学习直接作用于二维测量的左右两层可训练矩阵。这可以从数学上表示

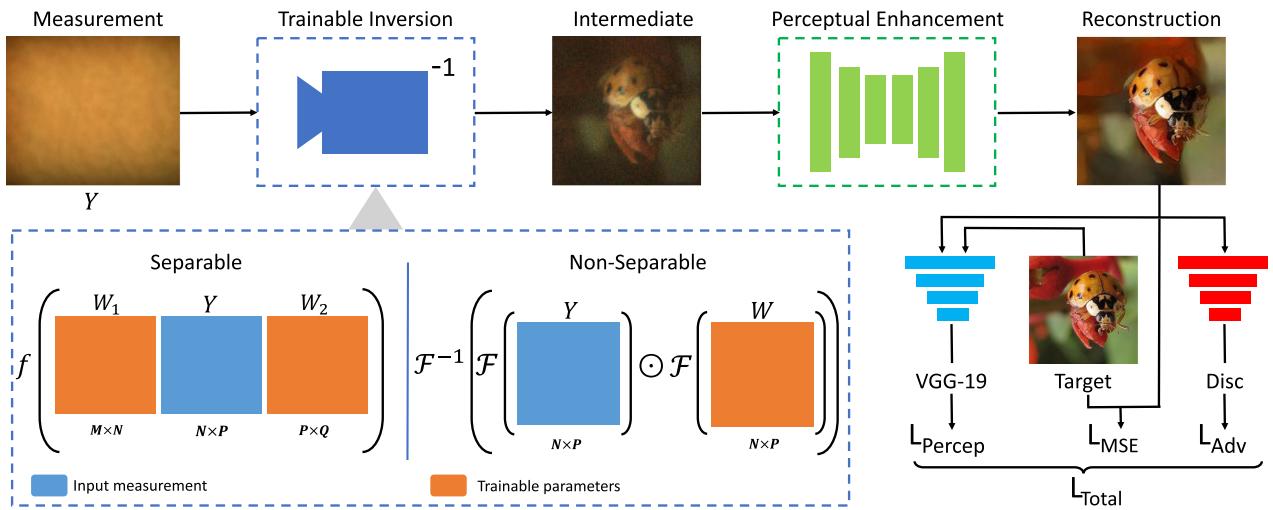


图 2.FlatNet 的总体架构。首先使用可训练相机反演层将无透镜相机测量映射到中间图像空间。该阶段针对可分离和不可分离的情况分别实现。然后, U-Net [31] 增强了中间重建的感知质量。我们在训练网络时使用三种损失的加权组合: 使用 aVGG16 网络 [33] 的感知损失 [32]、均方误差 (MSE) 和使用判别器神经网络 [34] 的对抗性损失。

表示为

$$X_{\text{interm}} = f(W_1 Y W_2), \quad (5)$$

其中  $X_{\text{interm}}$  是该阶段的输出,  $f$  是逐点非线性 (在我们的例子中是泄漏 ReLU),  $Y$  是输入测量,  $W_1$  和  $W_2$  是该阶段相应的权重矩阵。权重矩阵的维数取决于测量的维数和我们想要恢复的场景维数, 即  $W_1$  的维数与  $F_L$  转置的维数相同, 而  $W_2$  的维数与  $F_L$  的转置维数相同的  $F_R$ 。最终, 这些矩阵学会反转前向矩阵  $F_L$  和  $F_R$ 。我们将用于可分离无透镜模型的 FlatNet 版本称为 FlatNet-sep。正确初始化此阶段的权重矩阵非常重要, 这样网络就不会陷入局部最小值。这可以通过两种方式完成。校准初始化。对于这种方法, 我们用  $F_L$  的转置初始化权重矩阵  $W_1$ , 用  $F_R$  初始化  $W_2$ , 类似于反投影。(2)中的这些校准矩阵( $F_L$  和  $F_R$ )是通过[1]中描述的方法物理获得的。这种初始化模式可以在训练时加快收敛速度。未校准的初始化。FlatCam 的校准需要与显示监视器仔细对齐 [1], 这可能是一个耗时且不方便的过程, 尤其是对于大量的 FlatCam。即使校准中的微小误差也可能导致重建算法的性能严重下降。为了克服校准中涉及的问题, 我们还提出了一种免校准方法, 通过精心设计的伪随机矩阵来初始化权重矩阵。使用任何适当大小的伪随机矩阵进行初始化不会产生成功的重建。为了仔细设计随机初始化, 我们对 FlatCam 正向模型进行以下两个观察: 校准矩阵具有“类托普利茨”结构, 并且“类托普利茨”结构中常数项的斜率可以近似确定使用平面凸轮几何形状, 特别是掩模之间的距离

以及传感器和像素间距。由于 FlatCam 的几何结构是先验已知的, 我们可以构造具有适当斜率和大小的伪随机“类托普利茨”矩阵, 从而使我们的方法无需校准。我们在补充材料中更详细地讨论了这些伪随机矩阵的生成, 可以在计算机协会数字图书馆 <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2020.3033882> 上找到。权重矩阵  $W_1$  初始化为对应于  $F_L$  构造的随机矩阵的伴随, 而矩阵  $W_2$  用对应于  $F_R$  构造的随机矩阵初始化。我们观察到, 与转置初始化相比, 此初始化的训练时间略有增加。

### 3.1.2 Non-Separable Model

与可分离模型不同, 由于  $F$  的维数极大, 不可分离模型中的可训练反演阶段无法实现为矩阵乘法层。但是, 仍然可以在傅里叶域中实现。为了有效地实现反演阶段, 我们分析了方程 (1) 和 (3) 中给出的前向模型。根据观察, 前向模型对于适当的传感器尺寸是纯卷积的, 即前向运算由方程 (3) 描述, 我们以傅立叶域中哈达玛积的形式实现的学习逆的形式对不可分离情况的可训练反演阶段进行建模。这是因为方程 (3) 给出的循环系统的逆也是循环的, 并且可以通过傅立叶变换对角化。数学上, 该运算给出为

$$X_{\text{interm}} = \mathcal{F}^{-1}(\mathcal{F}(W) \odot \mathcal{F}(Y)), \quad (6)$$

其中  $X_{\text{interm}}$  是该阶段的输出,  $Y$  是测量值,  $\mathcal{F}$  和  $\mathcal{F}^{-1}$  是 DFT 和逆 DFT 运算,  $W$  是学习到的滤波器 (类似于  $W_1$  和  $W_2$  in 可分离模型) 并指的是 Hadamard 产品。

对于  $N \times M$  维测量,  $W$  的维度为  $N \times M$ 。我们发现使用非线性(例如 ReLU)对最终输出没有明显影响, 因此我们没有将其包含在不可分离模型中。等式(3)的卷积模型需要大型传感器, 因为无透镜系统中的 PSF 具有很大的空间维度, 并且在某些情况下使用如此大的传感器是不可行的。这种情况需要无透镜模型遵循方程(4)。当然, 我们无法通过卷积滤波器准确地表示方程(4)所描述的系统的逆, 因为该系统不再是循环的。因此, 人们可能会问, 如果使用更小的传感器, 所提出的可训练反转阶段是否仍然有效? 为了回答这个问题, 我们在第 4.4.2 节中展示, 通过对等式(6)中描述的可训练反演阶段进行少量修改, 我们可以处理这些裁剪卷积或非循环情况, 而不会显着降低性能。我们将这个用于不可分离无透镜模型的 FlatNet 版本称为 FlatNet-Gen.Calibrate Initialization。与可分离模型一样,  $W$  的初始化对于训练过程的收敛非常重要。假设我们有一个校准的 PSF 并且  $H$  是该 PSF 的傅立叶变换, 在我们的实验中, 我们使用  $F \xrightarrow{A} H \xrightarrow{K} F$  初始化  $W$ , 即 PSF 的正则化伪逆或众所周知的维纳滤波器。在此表达式中,  $K$  是正则化参数。未校准初始化。我们还提出了一种不需要显式 PSF 校准的初始化方案。给定掩模图案和相机几何形状, 我们可以模拟无透镜系统的 PSF。具体来说, 对于 PhlatCam, 给定掩模的高度轮廓, 我们使用菲涅尔传播来模拟 PSF, 如[2]中所述。这种初始化方案对于 PSF 超过传感器大小的情况特别有用(参见第 4.4.2 节)。这里应该注意的是, 这种初始化模式可用于我们可以访问高度轮廓的情况, 例如[2]中。对于无法粗略估计高度轮廓的情况, 例如使用随机扩散器时, 应首选初始化的校准模式。

### 3.2 感知增强

一旦我们获得可训练反演阶段的输出, 该输出与我们想要恢复的自然图像具有相同的维度, 我们就使用全卷积网络将其映射到感知增强的图像。由于其在图像到图像转换问题上的大规模成功及其多分辨率结构, 我们选择 U-Net [31] 将中间重建映射到最终的感知增强图像。我们将内核大小固定为  $3 \times 3$ , 而编码器中的滤波器数量逐渐从 128 个增加到 1024 个, 然后在解码器中减少回 128 个。最后, 我们将信号映射回 3 个 RGB 通道。对于不可分离的情况, 我们处理尺寸稍大的场景。与[35]类似, 我们发现在 U-Net 之前使用 Pixel-Shuffle[36]对中间图像进行下采样很有用。通过允许 U-Net 在较小的空间分辨率(因此更大的上下文区域)上运行, 我们可以增加的图像尺寸恢复更精细的细节。此外, Pixel-Shuffle 下采样不会丢弃像素, 因此可以与其他下采样方法完全反转。

### 3.3 判别器架构

我们在对抗性设置中训练 FlatNet-sep 和 FlatNet-gen。我们使用鉴别器框架将 FlatNet 的输出分类为真或假。我们发现使用鉴别器网络可以提高重建的感知质量。我们在判别器中使用 4 层 2 步长卷积, 然后进行批量归一化和 swish 激活函数[37]。FlatNet-sep 和 FlatNet-gen 使用相同的判别器架构。

### 3.4 损失函数

需要适当的损失函数来优化我们的系统以提供所需的输出。平均绝对误差(MAE)或均方误差(MSE)等像素损失已成功用于捕获信号失真。但是, 它们无法捕获图像的感知质量。由于我们的目标是通过无透镜测量获得高质量的真实感重建, 因此感知质量很重要。因此, 我们使用信号失真和感知损失的加权组合。我们的模型使用的损失如下: 均方误差。我们使用 MSE 来测量真实值和估计输出之间的失真。给定真实值图像  $I_{\text{true}}$  和估计图像  $I_{\text{est}}$ , 这给出为

$$\mathcal{L}_{\text{MSE}} = \|I_{\text{true}} - I_{\text{est}}\|_2^2 \quad (7)$$

知觉损失。为了测量估计输出和真实值之间的语义差异, 我们使用[32]中引入的感知损失。我们使用预先训练的 VGG-16 [33] 模型来处理感知损失。我们提取第二个卷积(激活后)和第二个最大池层之间以及第三个卷积(激活后)和第四个最大池层之间的特征图。我们分别将这些激活称为  $f_{22}$  和  $f_{43}$ 。该损失由下式给出:

$$\begin{aligned} \mathcal{L}_{\text{percept}} = & \|\phi_{22}(I_{\text{true}}) - \phi_{22}(I_{\text{est}})\|_2^2 \\ & + \|\phi_{43}(I_{\text{true}}) - \phi_{43}(I_{\text{est}})\|_2^2. \end{aligned} \quad (8)$$

对抗性损失。添加对抗性损失[34]、[38]是为了进一步使重建输出的分布接近真实图像的分布。给定第 3.3 节中描述的判别器  $D$ , 该损失给出为

$$\mathcal{L}_{\text{adv}} = \log D(I_{\text{est}}) \quad (9)$$

我们的判别器由 4 层 2 步长卷积组成, 后跟批量归一化和 ReLU 激活函数, 将生成器输出分类为真或假。生成器总损失。我们训练时 FlatNet 的总损失是三个损失的加权组合, 如下所示

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{MSE}} + \lambda_2 \mathcal{L}_{\text{percept}} + \lambda_3 \mathcal{L}_{\text{adv}}, \quad (10)$$

其中,  $\lambda_1$ 、 $\lambda_2$  和  $\lambda_3$  是分配给每个损失的权重。鉴别器损失。给定  $I_{\text{est}}$ 、 $I_{\text{true}}$  和判别器  $D$ , 判别器使用以下损失进行训练:

$$\mathcal{L}_{\text{D}} = \log D(I_{\text{true}}) + \log (1 - D(I_{\text{est}})) \quad (11)$$

语境损失。为了在未对齐的PhlatCam和网络摄像头对上微调FlatNet-gen（第4.5节中描述），我们仅使用[39]中提出的上下文损失。将输出图像特征( $f_{44} \delta I_{\text{est}} P$ )表示为 $f_{pq} g N_i l$ ，目标图像特征( $f_{44} \delta I_{\text{true}} P$ )表示为 $f_{qj} g N_j l$ ，并将每个特征图中的像素数表示为 $N$ ，上下文损失找到最近邻特征匹配 $q^{\frac{1}{4}} \arg \min q D \delta p$ ；对于每个 $p$ ,  $q \in P$ ,  $N_j = 1$ 。然后我们最小化所有这些特征对的总距离。我们这里采用的距离度量是余弦距离，尽管它也可以是 L1、L2 等。该损失项由下式给出：

$$L_{\text{上下文}} = \frac{1}{N} \sum_{i=1}^N \min_{j \in [N]} D(p_i, q_j). \quad (12)$$

基于计算成本和重建清晰度，我们发现  $f_{44}$  是合适的特征提取。on

#### 4 实验及结果

在本节中，我们将描述我们所有的实验。我们在真实数据上进行所有实验。我们将可分离模型的 FlatNet 称为 FlatNet-sep，将不可分离模型的 FlatNet 称为 FlatNet-gen。它们还将带有 -C 和 -UC 后缀，分别表示校准或未校准的初始化方法。除非特别提及，否则简单地使用 FlatNet-gen 或 FlatNet-sep 将表示 FlatNet-gen-C 或 FlatNet-sep-C，即使用校准的初始化方法初始化的 FlatNet。

##### 4.1 数据集

深度神经网络的监督训练需要大规模的标记数据集。然而，收集无镜头图像的大规模数据集是一项具有挑战性的任务。人们可以使用已知的无透镜模型来模拟可用自然图像数据集的测量结果。然而，由于一些非理想因素，这有时无法模拟真实的成像模型。为了克服这一挑战，我们通过将图像投影到显示器上并使用无镜头相机捕获该投影来收集大型数据集。这不仅可以处理无镜头相机的真实成像模型，还可以帮助我们收集无镜头图像的标记数据集。我们对 FlatCam [1] 和 PhlatCam [2] 遵循相同的数据集收集过程。对于我们的工作，我们使用 ILSVRC 2012[40] 的子集。具体来说，我们使用每个类别的 10 张随机图像作为我们的基本事实。在 1000 个班级中，我们保留了 990 个班级用于培训，其余的用于测试。因此，我们总共使用了 9,900 张图像进行训练，使用 100 张图像进行测试。在捕获数据集之前，我们调整显示器上显示的图像大小，以覆盖相机的整个视野 (FoV)。我们将此数据集称为“显示捕获数据集”。对于该数据集，地面实况图像是投影在监视器屏幕上的图像。监视器保持在相机的超焦距之外，以避免 PSF 随深度变化。FlatCam 原型的超焦距约为 1 英尺，PhlatCam 原型的超焦距约为 16 英寸。为了在真实场景上测试 FlatNet，我们还捕获直接放置在相机前面的物体的测量结果。使用 FlatCam，我们收集 15 个此类测量值，而使用 PhlatCam，我们收集 20 个此类测量值。我们将此数据集称为“直接捕获数据集”。该数据集没有相应的测量基础事实。到

为了证明 FlatNet-gen 在不受约束的室内场景中的有效性，我们使用图 1 中描述的设置收集了未对齐的 PhlatCam 和网络摄像头捕获的数据集。13. 该数据集由 475 个训练样本和 25 个测试样本组成。我们将此数据集称为无约束室内数据集。我们数据集中的样本如图 3 所示。代码和数据集已提供。<sup>1</sup>

#### 4.2 实施细节

FlatCam 原型采用 Point Grey Flea3 相机，配备 1.3MP e2v EV76C560 CMOS 传感器，像素尺寸为 5.3 毫米。当 FlatCam 被校准以产生  $256 \times 256$  输出图像时，所有地面真实图像的大小都调整为  $256 \times 256$ 。这确保了输入和地面实况对之间没有错位。我们直接使用拜耳测量，分为 4 个通道 (R, Gr, Gb, B)，作为网络的输入，并在网络内将它们转换为 3 通道 RGB。FlatCam 对尺寸  $512 \times 640 \times 4$  的测量（每批 4 个）用作训练的输入。由于内存限制，使用了较小的批量大小。我们将 1 设置为 1，将 2 设置为 1.2，将 3 设置为 0.6。对于转置初始化，我们训练模型进行 45 K 次迭代，而对于随机初始化，我们训练模型进行 60 次迭代。Adam [41] 优化器用于所有模型。我们从  $10^{-4}$  的学习率开始，每 5,000 次迭代逐渐将其减少一半。使用的 PhlatCam 原型是 Basler Ace4024-29uc，配备 12.2 MP SonyIMX226 传感器，像素尺寸为 1.85 mm。所有真实图像的大小都调整为  $384 \times 384$ ，等于原型的 FoV。我们直接使用拜耳测量，分为 4 个通道 (R, Gr, Gb, B)，作为网络的输入，并在网络内将它们转换为 3 通道 RGB。我们使用与 FlatNet-sep 相同的 i 集。使用的完整测量尺寸为  $1280 \times 1408 \times 4$ 。对于第 4.4.2 节的小型传感器实验，我们使用尺寸为  $608 \times 864 \times 4$  的测量。

#### 4.3 与其他方法的比较 4.3.1 可分离无镜头模型

在本小节中，我们展示了遵循可分离模型的振幅 maskFlatCam 的结果。我们将 FlatNet-sep 与 [1] 中描述的封闭形式 Tikhonov 重建以及使用 TVAL3 [19] 实现的基于总变分的重建进行比较。定性讨论。在图 4 中，我们将我们的方法（具有未校准初始化的 FlatNet-sep-UC 和具有校准初始化的 FlatNet-sep-C）与传统方法（Tikhonov 和 TVAL3）进行了比较。从重建中可以看出，Tikhonov 正则化重建容易出现严重的渐晕效应，这种效应在 TVAL3 结果中有所减弱。图 4 中的插图显示了我们的方法保留了更精细的细节。图 5 显示了直接捕获测量的各种方法的性能。吉洪诺夫正则化有抑制低信号值的趋势，因此很难恢复图 5 中大多数场景的照明不良的背景。TVAL3 [19] 的情况也类似。另一方面，FlatNet-sep 可以产生更高质量的真实感重建。

<sup>1</sup> Link: <https://github.com/siddiquesalman/flatnet>

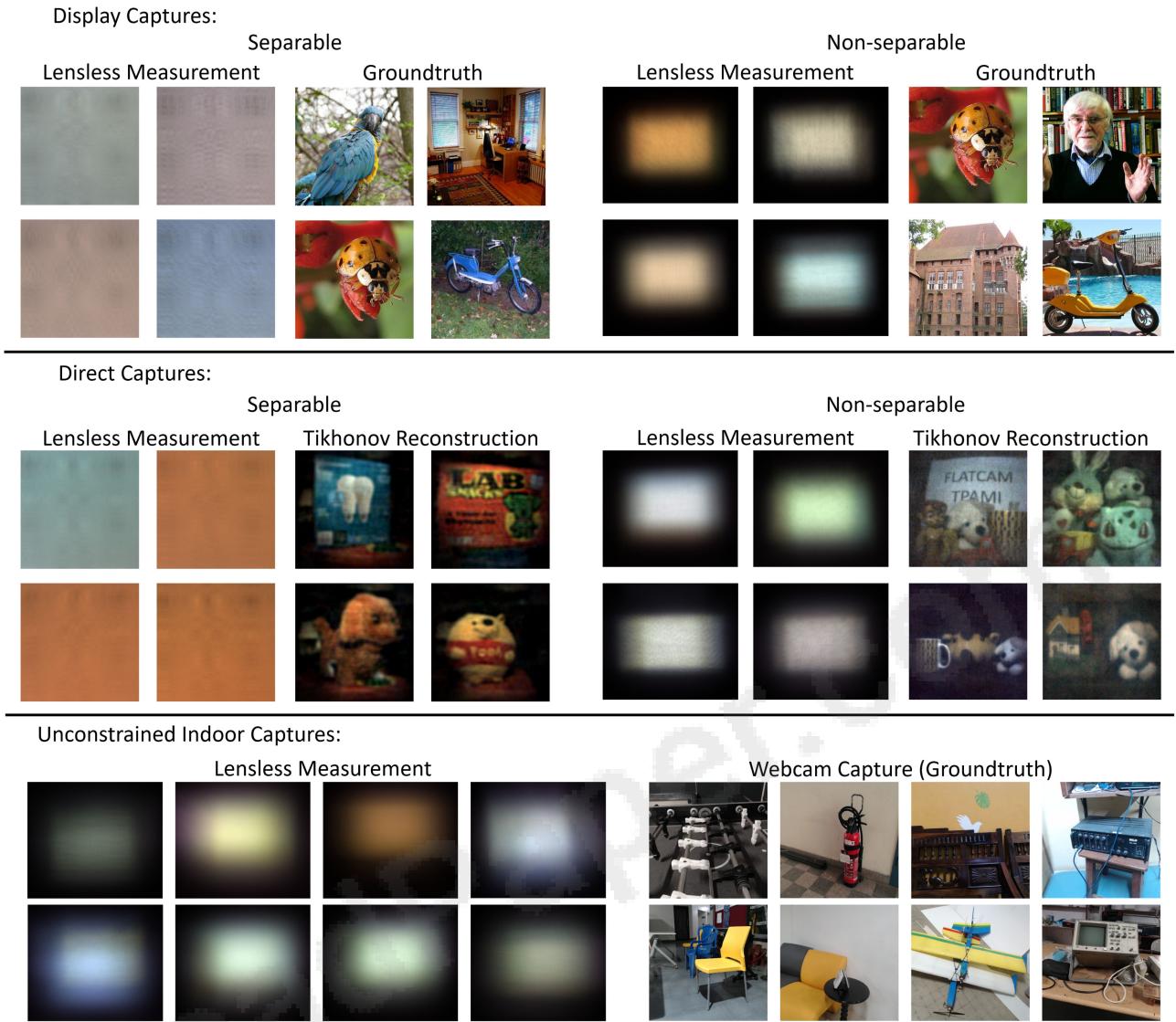


图 3. 我们收集的数据集中的样本。我们所有的实验都是基于使用无透镜原型捕获的真实数据进行的。我们使用可分离和不可分离原型收集 DisplayCaptured 数据集, 分别训练 FlatNet-sep 和 FlatNet-gen。我们还通过将物体放置在受控照明下的无镜头相机前面来收集直接捕获数据集。最后, 为了提高 FlatNet 的鲁棒性, 我们使用 PhlatCam 和网络摄像头对收集了无约束室内场景的数据集。

请注意, 我们的未校准模型 FlatNet-sep-UC 提供了与校准模型 FlatNet-sep-C 相似的性能。因此, 与其他方法不同, 我们的方法不需要显式校准。定量讨论。我们在表 1 中展示了可分离掩模 FlatCam 的 FlatNet 的定量性能。我们使用 PSNR、SSIM 和最近提出的 LPIPS [42] 进行评估。较高的 PSNR 和 SSIM 分数表示更好的性能, 而较低的 LPIPS 表示更好的感知质量。可以清楚地看到, 我们使用转置初始化 (FlatNet-sep-C) 的方法优于 FlatCam 的所有其他重建技术。下一个最佳方法是使用随机初始化的 FlatNet-sep (FlatNet-sep-UC), 与其他方法不同, 它是一种免校准技术。我们还在同一个表中比较了各种方法的推理时间。Tikhonov 和 TVAL3 [19] 正则化重建在具有 16GB RAM 的 Intel Core i7 CPU 上计算, 而其余方法在 Nvidia GTX 1080 Ti GPU 上评估。

#### 4.3.2 Non-Separable Lensless Model

对于不可分离模型的实验, 我们将 FlatNet-gen 与传统方法和基于学习的方法进行比较。我们在下面描述这些方法。传统方法。在传统方法中, 我们将 FlatNet-gen 与在傅立叶域中实现的传统 Tikhonov 正则化重建 (如维纳恢复滤波器) 和使用 ADMM [4] 实现的全变分正则化重建进行比较。基于学习的方法。对于基于学习的方法, 我们使用 [28] 中描述的 展开深度网络。然而, 为了公平起见, 我们使用五阶段展开的 ADMM, 然后是感知增强阶段。定性讨论。图 6 显示了 PhlatCam 的显示捕获重建。与传统的吉洪诺夫正则化重建或维纳反卷积和基于 ADMM 的方法相比, 我们可以清楚地看到 FlatNet-gen 具有更高质量的重建。与 Le-ADMM 模型相比, 它还可以实现更好的质量重建。这种趋势在

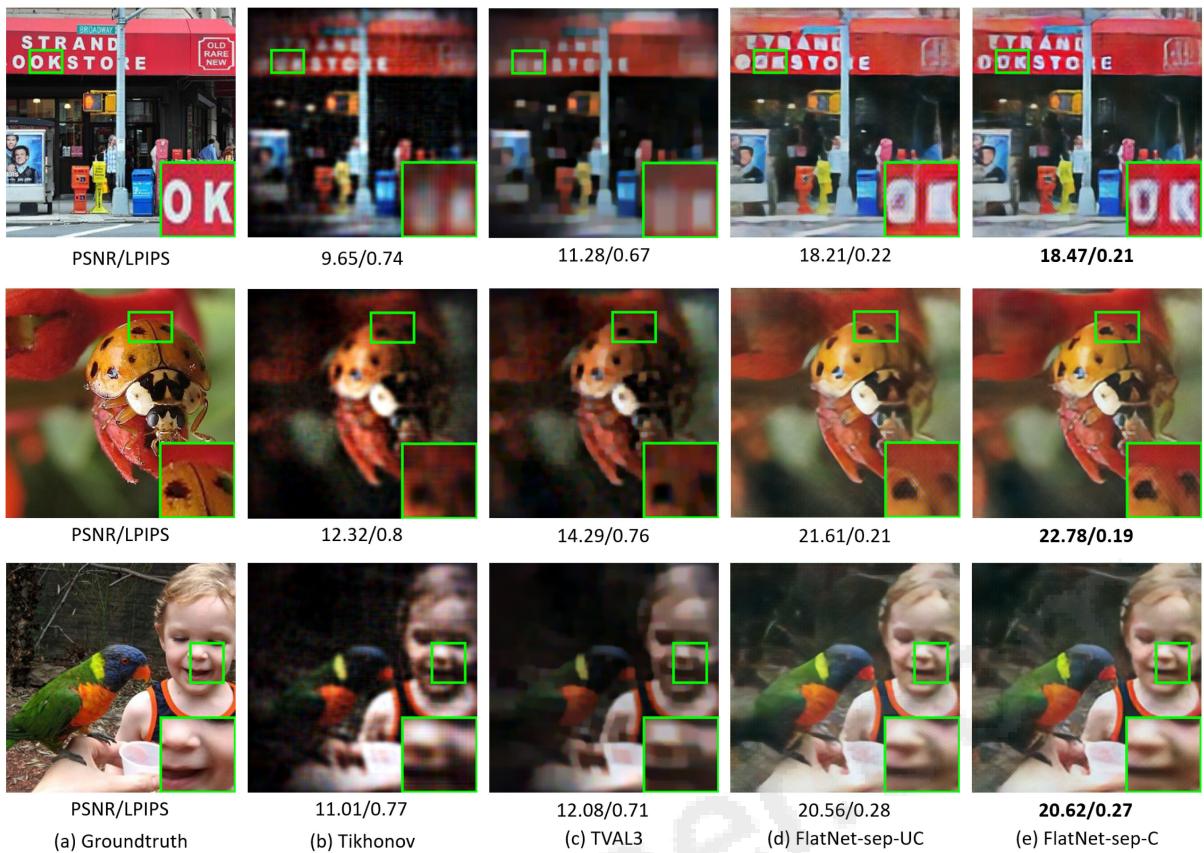


图 4. 显示 FlatCam 捕获的重建。地面实况图像如 a) 所示。更精细的细节（例如第一幅图像中的文本和第二幅图像中昆虫上的斑点）在 b) Tikhonov 正则化和 c) TVAL3 重建中丢失。对于 d) 未校准和 e) 校准初始化，FlatNet-sep 中可以更好地保留更精细的细节。



图 5. FlatCam 的直接捕获重建。a) 边界和较暗区域的细节在吉洪诺夫正则化重建中丢失。b) TVAL3 重建了边界但无法恢复清晰度。所提出的 c) 随机和 d) 转置初始化的端到端模型可产生最佳重建。这些方法对噪声具有鲁棒性，并且不包含任何正则化参数。

表 1 显示捕获的 FlatCam 测量值的平均指标

| Method         | PSNR (in dB) | SSIM | LPIPS | Inference Time (in sec) |
|----------------|--------------|------|-------|-------------------------|
| Tikhonov       | 10.95        | 0.33 | 0.795 | 0.03                    |
| TVAL3          | 11.81        | 0.36 | 0.752 | 45.28                   |
| FlatNet-sep-UC | 19.06        | 0.62 | 0.274 | 0.006                   |
| FlatNet-sep-C  | 19.62        | 0.64 | 0.256 | 0.006                   |

FlatNet-sep with transpose initialization (FlatNet-sep-C) gives the best result. Comparable performance of FlatNet-sep-UC indicates that our approach can be used for situations where careful calibration isn't possible.

在图 7 中的直接捕获重建中也观察到了性能。还应该注意的是，尽管 Le-ADMM 的参数较少，但由于存在大量中间/原始和对偶变量，因此内存和计算量非常大在展开的 ADMM 的每个阶段进行计算。正是由于内存消耗的显着增加，它变得不可行。

在捕获的 PhlatCam 测量结果上实现此模型，无需下采样。在我们的比较中，我们将测量值下采样 4 倍（类似于 [28]），然后再将它们传递给 Le-ADMM 网络。除非明确提及，否则我们将这种下采样 Le-ADMM 模型称为 Le-ADMM。下采样操作会导致重建分辨率的妥协，导致最终重建中观察到的清晰度缺乏。另一方面，FlatNet-gen 的内存要求显著降低，不需要任何下采样预处理，从而防止任何清晰度或分辨率的损失。我们还在补充材料中提供了使用未校准 PSF 初始化的 FlatNet-gen 的比较，可在线获取。我们将此模型称为 FlatNet-gen-UC. Quantitative Discussion。表 2 提供了定量结果。除了未校准的 FlatNet-gen 模型之外，我们还在该表中提供了未校准版本的 Le-ADMM 的性能。它被称为 Le-ADMM-

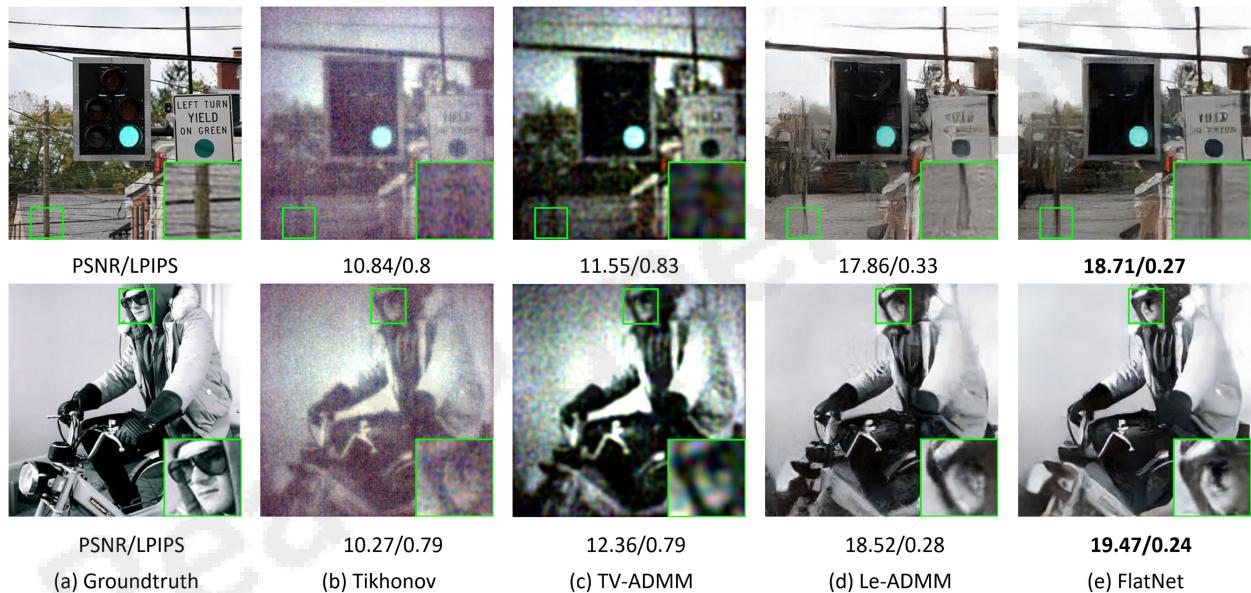


图 6. 显示 PhlatCam 捕获的重建图像。虽然基于学习的方法明显优于 Tikhonov 和基于 TV 的 ADMM 等传统方法，但 FlatNet-gen 在重建更精细的细节方面具有卓越的性能。



图 7. PhlatCam 的直接捕获重建。FlatNet-gen 的伪影较少，而 Le-ADMM 则存在模糊重建和幻觉伪影的问题。

表 2 显示捕获的 PhlatCam 测量值的平均指标

| Method         | PSNR (in dB) | SSIM        | LPIPS        | Inference Time (in sec) |
|----------------|--------------|-------------|--------------|-------------------------|
| Tikhonov       | 12.67        | 0.25        | 0.758        | 0.03                    |
| TV-ADMM        | 13.51        | 0.26        | 0.755        | 180                     |
| Le-ADMM-UC     | 18.35        | 0.49        | 0.407        | 0.08                    |
| Le-ADMM-C      | 20.29        | 0.51        | 0.333        | 0.08                    |
| FlatNet-gen-UC | 20.53        | 0.54        | 0.318        | 0.03                    |
| FlatNet-gen-C  | <b>20.94</b> | <b>0.55</b> | <b>0.296</b> | <b>0.03</b>             |

FlatNet-gen produces higher quality results without compromising on the inference time for both the real PSF case (FlatNet-gen-C) and the simulated PSF case (FlatNet-gen-UC). Le-ADMM shows larger difference in quality between the real and simulated PSF cases owing to its stronger dependence on the PSF.

加州大学。定量指标保持与视觉结果的一致性。可以清楚地看到, FlatNet-gen 在数量上优于所有其他方法。FlatNet-gen-UC 的性能几乎与 FlatNet-gen-C 相当, 并且优于 Le-ADMM-UC。值得注意的是, 与 Le-ADMM-C 和 Le-ADMM-UC 相比, FlatNet-gen-C 和 FlatNet-gen-UC 之间的差异较小。这主要是由于 Le-ADMM 对真实 PSF 的依赖性更强, 而 FlatNet-gen 仅需要 PSF 知识才能更好地初始化, 并在训练后学会收敛到更好的逆。我们还提供了比较方法的运行时。对于 Wiener 和基于 TV 的 ADMM, 我们报告 CPU 上的速度, 而对于其他模型, 我们报告 GPU 中前向传递的速度。假设真实测量的尺寸为  $1280 \times 1408$ , 我们另外将 FlatNet-gen 的可训练反转阶段与展开的阶段进行比较 Le-ADMM 的 ADMM 块 (不带 U-Net) 在内存和计算方面如表 3 所示。我们提供了内存消耗 (以兆字节为单位, 在 Nvidia GTX 1080 Ti GPU 上计算) 和计算量 (inFLOPs, 计算得出) 理论上) 使用这两种方法处理一张图像所需的。我们展开 ADMM 进行 5 次迭代。在表中, Le-ADMM-Full 指的是展开的 ADMM, 没有任何下采样, 而 Le-ADMM-Downsampled 指的是 PSF 和场景下采样 4 倍的情况。可以看出, 完整分辨率的 Le-ADMM 需要大量内存, 如果考虑部署, 这将产生负面影响。此外, 附加了像 U-Net 这样的密集 CNN, Le-ADMM-Full 很难在传统的 GPU 上实现, 因此需要对测量结果进行下采样, 这又会导致重建质量的下降。还应该注意在展开的 ADMM 块中针对 PSF 和场景的特定维度执行的计算量。由于一系列中间估计取决于

TABLE 3  
Memory and FLOP Comparison

| Method              | Memory (in MB) | Computation (in MFLOP) |
|---------------------|----------------|------------------------|
| Le-ADMM-Full        | 6300           | 1290                   |
| Le-ADMM-Downsampled | 1000           | 65                     |
| FlatNet-gen         | <b>990</b>     | <b>53</b>              |

Le-ADMM (完整和 4 倍下采样版本) 中 ADMM 块的五次展开迭代的内存消耗和 FLOP 与我们提出的 FlatNet-gen 的可训练反转阶段的比较。



图 8. FlatNet 与 Tikh+U-Net 的比较。顶行显示了 FlatNet-sep 与 Tikh+U-Net 的比较, 底行显示了 FlatNet-gen 与 Tikh+U-Net 的比较。与 Tikh+U-Net 相比, FlatNet 对于可分离和不可分离模型提供了更清晰、更真实的重建。

在傅里叶变换和傅里叶逆变换上, Le-ADMM-Full 的计算量会大幅增加。FlatNet-gen 为分辨率、内存和计算要求提供了更好的权衡, 这对于无透镜系统至关重要, 而无透镜系统在设计上会受到重建分辨率较差的影响。

#### 4.4 进一步分析 4.4.1 学习反转阶段的效果

在本节中, 我们强调 FlatNet 端到端学习策略的重要性。我们将 FlatNet 与仅具有感知增强块的网络进行比较。我们用吉洪诺夫正则化重建来训练这个网络。为了训练这个网络, 我们使用等式 (10) 中定义的相同损失。我们称这种方法为 Tikh+U-Net。我们对可分离和不可分离的无透镜模型实施这种方法。图 8 的顶行比较了 FlatNet-sep 与 Tikh+U-Net 的重建质量。我们可以很容易地观察到与 Tikh+U-Net 相比, FlatNet-sep 获得的重建质量有所提高。Tikh+U-Net 的重建结果更加模糊, 伪影被放大。我们还在图 8 的底行中比较了 FlatNet-gen 与其相应的 Tikh+U-Net 的性能。FlatNet-gen 提供了比 Tikh+U-Net 更清晰的重建。表 4 为上述分析提供了定量分析-姐姐。我们可以看到 FlatNet 的性能优于 Tikh+U-Net

表 4 FlatNet 与 Tikh+U-Net 的比较

| Methods                    | PSNR (in dB) | LPIPS        |
|----------------------------|--------------|--------------|
| <b>Separable Model</b>     |              |              |
| Tikh+U-Net                 | 18.90        | 0.322        |
| <b>FlatNet</b>             | <b>19.62</b> | <b>0.256</b> |
| <b>Non-separable Model</b> |              |              |
| Tikh+U-Net                 | 20.60        | 0.298        |
| <b>FlatNet</b>             | <b>20.94</b> | <b>0.296</b> |

上半部分将 FlatNet-sep 与 Tikh+U-Net 用于可分离无透镜模型进行比较, 下半部分将 FlatNet-gen 与相应的 Tikh+U-Net 进行比较。对于可分离和不可分离模型, FlatNet 的性能均优于 Tikh+U-Net, 因为它学习端到端映射。

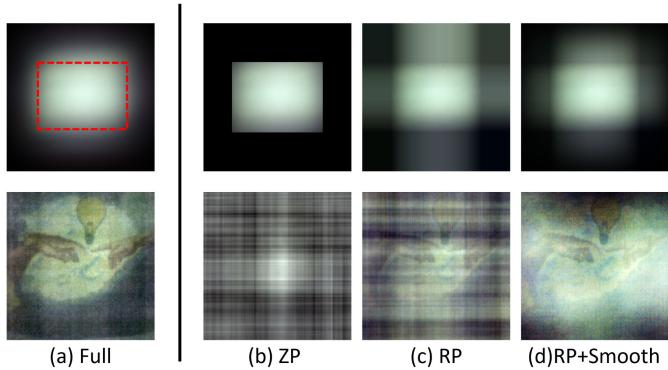


图 9. 裁剪测量中填充对维纳反卷积的影响。顶行显示测量结果，而底行显示相应的维纳重建。(a) 全面测量。红色框表示裁剪掉的区域。(b) 补零测量和相应的重建。(c) 重复填充测量和相应的重建。(d) 平滑的复制填充测量以及相应的重建。本工作中使用的(d)中的线伪影显着减少。

在 PSNR 和 LPIPS 方面，可分离和不可分离模型。人们可能会注意到，FlatNet-gen 和 Tikh+U-Net 之间的差异并不像 FlatNet-sep 与其对应的 Tikh+U-Net 之间的差异那么显着。这是由于与 FlatCam 相比，PhlatCam 的 Tikhonov 重建质量更高 [2]。然而，应该注意的是，Tikh+U-Net 严格基于前向模型的卷积假设，并且当违反该假设时表现不佳，这将在第 4.4.2 节中进行验证。

#### 4.4.2 裁剪测量的性能

正如我们在第 2 节中已经看到的，如果传感器的尺寸与真实测量尺寸相比较小，则基于掩模的无镜头相机中的前向操作不再是卷积，即前向模型由方程 (4) 给出。这与大 PSF 相结合，使得无透镜重建对依赖循环或卷积假设（例如维纳反卷积）的传统重建方法提出了挑战。这自然会引出一个问题：

提出的 FlatNet-gen 的可训练反演层基于学习的傅立叶域反演，对于与循环假设的偏差显着的情况是否具有鲁棒性？换句话说，FlatNet-gen 是否能够处理由于传感器尺寸有限和光圈全开而导致大量像素被丢弃的测量结果？在本节中，我们展示了我们可以处理小传感器尺寸的情况，而不会在重建质量方面损失太多，并且比 Le-ADMM 表现更好，Le-ADMM 明确地尝试处理裁剪掉的像素。对于我们的实验，我们从 7 MP 全传感器测量中获取尺寸为  $608 \times 864$  的中心裁剪。实际上，这可以被认为是使用 2 MP 传感器而不是 7 MP 传感器。根据 [30] 中的观察，我们复制裁剪后的测量结果作为预处理步骤。为了平滑由于填充引起的不连续性，我们将这个填充测量值与高斯滤波框相乘。我们的填充方法的有效性可以在图 9 中观察到。从数学上讲，可训练的反转阶段变为

$$X_{\text{interm}} = \mathcal{F}^{-1}(\mathcal{F}(W) \odot \mathcal{F}(\text{pad}(Y))). \quad (13)$$

这是对方程 (6) 的修改，以考虑裁剪测量。 $\text{pad}\ddot{\circ}:\mathbb{P}$  指的是上面描述的填充和平滑操作。Tikh+U-Net 也遵循相同的填充和平滑程序应用于裁剪后的测量结果。图 10 显示了 Tikh+U-Net、Le-ADMM 和 FlatNet 的显示捕获裁剪测量与完整测量的重建质量。即使在填充测量值之后，维纳恢复图像中也存在无法使用 Tikh+U-Net 有效去除的伪影。Le-ADMM 的性能比 Tikh+U-Net 稍好，因为它的中间阶段可以近似估计未裁剪的测量值。然而，它的裁剪不如 FlatNet-gen 那样稳健。类似地，在图 11 中，我们显示了直接捕获的裁剪测量的重建。可以清楚地看到 Tikh+U-Net 和 Le-ADMM 存在明显的色彩伪影。然而，这些伪影在 FlatNet-gen 重建中并不重要。表 5 给出了显示捕获数据集上每个模型的平均分数的比较。

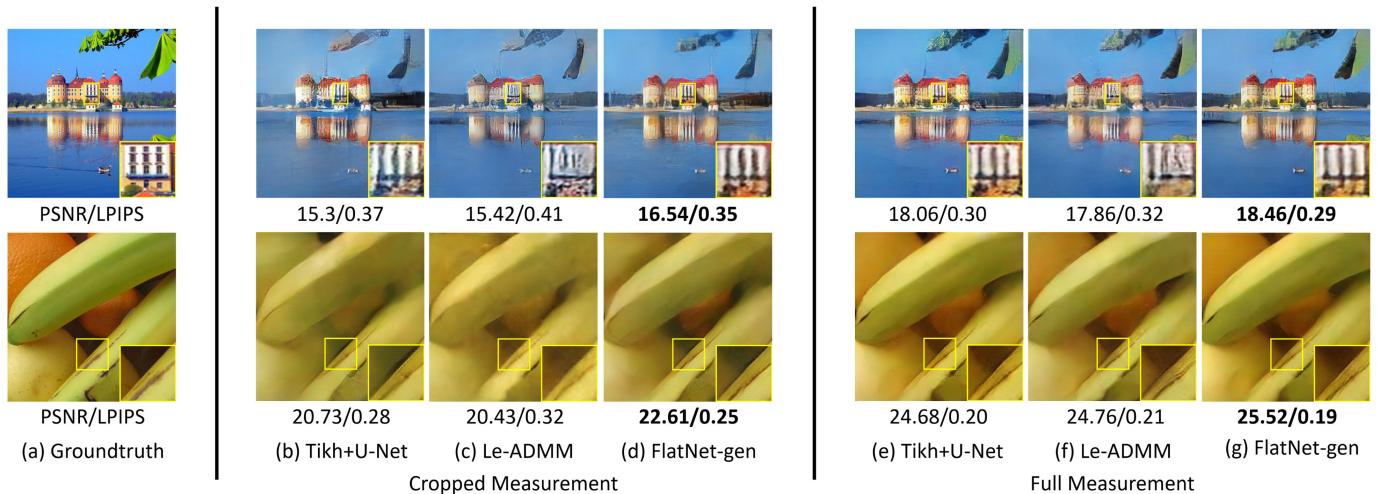


图 10. 显示裁剪后的 PhlatCam 测量结果的捕获重建。在 FlatNet 的裁剪测量和完整测量中观察到的性能差异很小。然而，对于 Le-ADMM 和 Tikh+U-Net 来说，这种差异都很大。

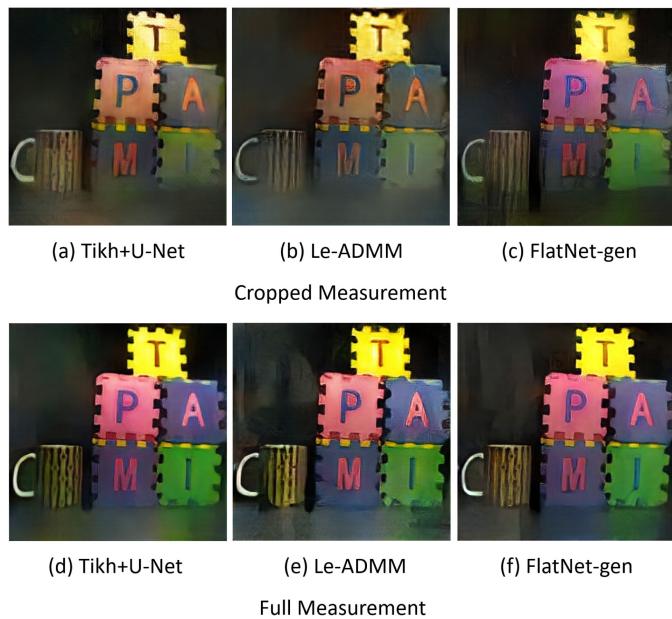


图 11. 裁剪后的 PhlatCam 测量的直接捕获重建。我们可以看到 FlatNet-gen 表现得相当好，而 Le-ADMM 和 Tikh+U-Net 都崩溃了。这可以通过 Tikh+U-Net 和 Le-ADMM 中的字母颜色和模糊外观（尤其是在边界周围）观察到。

应该注意的是，对于用于获得图的模型。如图 10 和 11 以及表 5 所示，PSF 大小 ( $608 \times 870$ ) 超过了假设的传感器大小 ( $606 \times 864$ )。在这种情况下，估计真实 PSF 是一个繁琐的过程，可以使用未校准的 FlatNet-gen-UC。从表 5 中，我们可以看到 FlatNet-gen 优于所有其他学习方法。FlatNet-gen-UC 与 FlatNet-gen 具有相当的性能，而 Tikh+U-Net-UC 和 Le-ADMM-UC 则出现故障：表明这些方法需要精确的 PSF 校准。补充材料中提供了用于裁剪测量的 FlatNet-gen-UC 的视觉结果，可在线获取。除了上面提到的裁剪尺寸之外，我们还在图 12 中展示了基于学习的方法针对各种不同裁剪尺寸的性能。, 我们相对于完整测量值标准化裁剪测量值的大小。可以看出，FlatNet-gen 在所有作物尺寸上始终优于 Le-ADMM 和 Tikh+U-Net。还应该指出的是，FlatNet-sep 在设计上对非循环场景具有鲁棒性，因为它涉及空间中的学习反转领域。

表 5 裁剪显示捕获的  
PhlatCam 测量值的平均指标

| Method         | PSNR(in dB)  | SSIM        | LPIPS        |
|----------------|--------------|-------------|--------------|
| Tikh+U-Net-UC  | 17.53        | 0.45        | 0.438        |
| Tikh+U-Net-C   | 18.34        | 0.48        | 0.376        |
| Le-ADMM-UC     | 17.94        | 0.45        | 0.410        |
| Le-ADMM-C      | 18.72        | 0.48        | 0.371        |
| FlatNet-gen-UC | 18.72        | 0.48        | 0.375        |
| FlatNet-gen-C  | <b>19.29</b> | <b>0.50</b> | <b>0.365</b> |

对于真实 (FlatNet-gen-C) 和模拟 PSF 案例 (FlatNet-gen-UC)，FlatNet-gen 的表现始终优于其他学习方法。值得注意的是，FlatNet-gen-UC 的性能与基于真实 PSF 的 Le-ADMM 一样好。

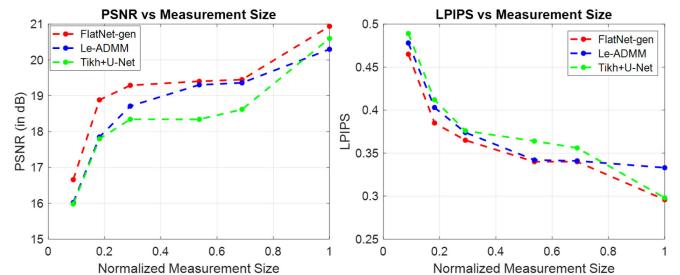


图 12. 基于学习的技术在不同数量作物上的表现。我们绘制了 FlatNet-gen、LeADMM 和 Tikh+U-Net 在相对于完整测量尺寸标准化的各种测量尺寸下的 PSNR 和 LPIPS。我们可以看到，对于所有作物尺寸，FlatNet-gen 始终优于其他基于学习的方法。

#### 4.5 无约束室内场景下的性能

在前面的部分中，我们使用在显示捕获的数据集上训练的 FlatNet 进行了所有实验。然而，在野外捕获的实际测量值与显示捕获的测量值不同，原因如下：a) 与显示捕获的测量值相比，现实世界捕获的噪声量明显更高，b) 在不受约束的设置中，明亮的场景点超出了主光线角 (CRA) 描述的 FoV 也会影响捕获的测量结果，而使用填充整个 CRA 定义的 FoV 的监视器捕获的显示器捕获测量结果则不然。为了考虑到这些差异并使我们的 FlatNet 对现实世界场景具有鲁棒性，我们使用我们捕获的名为“无约束室内数据集”的现实世界数据集对 FlatNet 进行微调。该数据集由使用图 13 中描述的设置收集的未对齐网络摄像头和 PhlatCam 捕获数据组成。我们收集了 500 对此类数据，保留 475 对用于训练，25 对用于测试。我们以较小的学习率对整个网络进行微调（可训练反转阶段为 10 A12，感知增强阶段为 10 A6）。为了解决 PhlatCam 和网络摄像头捕获之间的不对齐问题，我们仅使用之前针对未对齐数据提出的 ContextualLoss [39]。图 13 显示了我们的一些经过微调和未经微调的重建结果以及网络摄像头捕获以供参考。可以观察到，微调导致了更加真实的重建。在在线提供的补充材料中，我们展示了根据裁剪后的无约束室内测量进行的重建。

## 5 讨论与结论

在本文中，我们提出了一种称为 FlatNet 的端到端可训练深度网络，用于从无透镜测量中重建逼真的场景。尽管无透镜成像提供了许多前景，但它在某种程度上受到重建图像质量的限制。在本文中，我们试图弥合无镜头成像的前景与其性能之间的差距。FlatNet 利用前向模型的物理原理（通过可训练的相机反转）和数据驱动方法的成功来学习从高度复用的无镜头捕捉到估计场景的逼真映射。与基于展开的网络[28]不同，它具有内存和计算要求低的优点，这是独立设备的理想标准。我们还表明，通过对训练有素的 FlatNet 进行微调

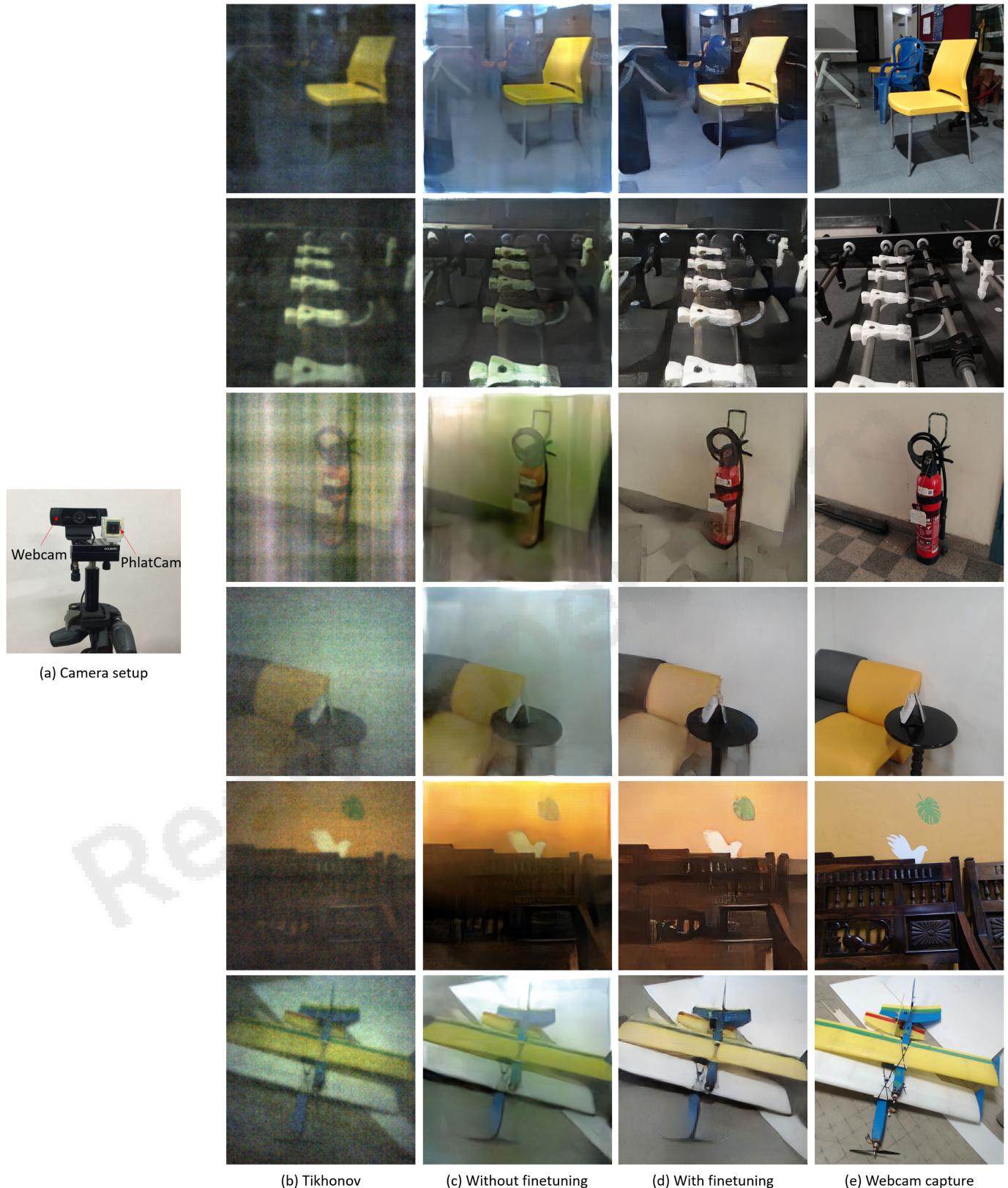


图 13. 无约束室内场景的真实感重建。(a) PhlatCam-Webcam 设置捕获用于微调 Flat-Net-gen 的数据集。(b) 吉洪诺夫重建。(c) 仅在显示捕获的数据上训练的 FlatNet-gen 的重建。(d) 使用 FlatNet-gen 在无约束的室内捕获上进行微调的重建。(e) 网络摄像头图像供参考。微调使重建更加真实。

使用未对齐的网络摄像头 PhlatCam 室内场景显示捕获的测量结果，我们可以使用这些超薄传感器在野外恢复逼真的图像。

还应该指出的是，与大多数基于 FAN 的方法一样，FlatNet 重建会受到幻觉伪影的影响，这种幻觉伪影更倾向于照片写实主义而不是高保真度。

因此, 当手头的任务对这些幻觉伪像(例如医学成像)至关重要时, 应谨慎使用 FlatNet。尽管如此, 在这样的关键系统中, 我们仍然可以使用 FlatNet 的可训练相机反演, 并对感知增强和损失进行适当的修改。将来, 研究掩模或PSF和重建算法的协同设计将是很有意思的基于 Formask 的无镜头相机。

## ACKNOWLEDGMENTS

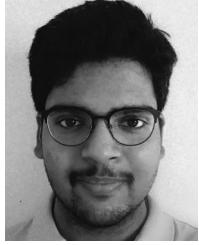
这项工作得到了 NSF CAREER: IIS-1652633、NSF EXPEDITIONS: CCF-1730574、DARPA NESD: HR0011-17-C0026、NIH Grant: R21EY029459 和 Qual-comm Innovation Fellowship 2020 India 的部分支持。Salman Siddique Khan 和 Varun Sundar 对这项工作做出了同样的贡献。

## REFERENCES

- [1] M. S. Asif、A. Ayremiou、A. Sankaranarayanan、A. Veeraraghavan 和 R. G. Baraniuk, “FlatCam: 使用编码孔径和计算的薄型无镜头相机”, IEEE Trans. 计算。图像, 卷。3, 没有。3, 第 384–397 页, 2017 年 9 月。[2] V. Boominathan、J. K. Adams、J. T. Robinson 和 A. Veeraraghavan, “PhlatCam: 设计基于相位掩模的薄型无透镜相机”, IEEE Trans. 模式肛门。马赫。智力, 卷。42, 没有。7, 第 1618–1629 页, 7 月。2020 年[3] A. W. Lohmann, “镜头系统的缩放定律”, Appl. 选项, 第 28 卷, 编号。23, 第 4996–4998 页, 1989 年。[4] N. Antipa 等人, “DiffuserCam: 无透镜单次曝光 3D 成像”, Optica, 卷。5, 没有。1, 第 1–9 页, 2018 年。[5] E. J. Tremblay、R. A. Stack、R. L. Morrison 和 J. E. Ford, “使用环形折叠光学器件的超薄相机”, Appl. 选项, 卷。46, 没有。第 4 页。463–471, 2007.[6] V. Boominathan 等人, “无透镜成像: 计算复兴”, IEEE 信号处理。杂志, 卷。33, 没有。5, 第 23–35 页, 9 月。2016.[7] S. S. Khan、A. V. R. V. Boominathan、J. Tan、A. Veeraraghavan 和 K. Mitra, “走向高度复用无透镜图像的真实感重建”, Proc. IEEE 国际。会议。计算。视觉, 2019, 第 7859–7868 页。[8] R. Dicke, “用于 X 射线和伽马射线的散孔相机”, 《天体物理学》。J., 卷。153, 1968, 艺术。不。L101.[9] E. Caroli、J. Stephen、G. Di Cocco、L. Natalucci 和 A. Spizzichino, “X 射线和伽马射线天文学中的编码孔径成像”, SpaceSci。修订版, 卷。45, 没有。3/4, 第 349–403 页, 1987 年。[10] G. Kim、K. Isaacson、R. Palmer 和 R. Menon, “仅使用图像传感器的无镜头摄影”, Appl. 选项, 卷。56, 没有。23 页。6450–6456, 2017.[11] D. G. Stork 和 P. R. Gill, “无透镜超微型 CMOS 计算成像器和传感器”, Proc. 国际。会议。传感器技术。应用, 2013 年, 第 186–190 页。[12] T. Shimano、Y. Nakamura、K. Tajima、M. Sao 和 T. Hoshizawa, “采用菲涅耳区孔径的无透镜光场成像: 准相干编码”, Appl. 选项, 卷。57, 没有。11, 第 2841–2850 页, 2018 年。[13] G. Huang、H. Jiang、K. Matthews 和 P. Wilford, “压缩传感的无透镜成像”, Proc. 中。IEEE 国际。会议。图像处理, 2013, 第 2101–2105 页。[14] G. Satat、M. Tancik 和 R. Raskar, “具有压缩超快传感功能的无透镜成像”, IEEE Trans. 计算。图像, 卷。3, 没有。第 3 页。398–407, 2017 年 9 月。[15] W. Chi 和 N. George, “具有相位编码孔径的光学成像”, Opt. 快报, 卷。19, 没有。5, 第 4294–4300 页, 2011 年。[16] M. J. DeWeert 和 B. P. Farm, “使用可分离双托普利茨掩模的无透镜编码孔径成像”, Opt. 工程, 卷。54, 没有。2, 2015, 艺术。不。023102.[17] M. F. Duarte 等人, “通过压缩采样进行单像素成像”, IEEE 信号处理。杂志, 卷。25, 没有。2, 第 83–91 页, 2008 年 3 月。[18] N. Antipa、P. Oare、E. Bostan、R. Ng 和 L. Waller, “静态照片视频: 使用卷帘快门的无镜头成像”, Proc. IEEE 国际会议。计算。摄影, 2019 年, 第 1–8 页。[19] C. Li、W. Yin、H. Jiang 和 Y. Zhang, “一种有效的增广拉格朗日方法及其在总变差最小化中的应用”, Comput. 最佳。应用, 卷。56, 没有。3, 第 507–530 页, 2013 年。[20] D. Reddy、A. Veeraraghavan 和 R. Chellappa, “P2C2: 用于高速成像的可编程像素压缩相机”, Proc. IEEE Conf. 计算。维斯。模式识别, 2011, 第 329–336 页。[21] A. Dave、A. K. Vadathya、R. Subramanyam、R. Baburajan 和 K. Mitra, “使用深度像素级先验解决逆计算成像问题”, IEEE Trans. 计算。图像, 卷。5, 没有。1, 页。2019 年 3 月 37 日至 51 日。[22] A. Dave、A. K. Vadathya 和 K. Mitra, “使用循环生成模型的压缩图像恢复”, Proc. 中。IEEE 国际会议。图像处理, 2017, 第 1702–1706 页。[23] J. H. R. 张、C.-L. Li、B. Poczos、B. V. K. Vijaya Kumar 和 A. C. Sankaranarayanan, “一个网络可以解决所有问题 - 使用深度投影模型解决线性逆问题”, Proc. IEEE Int. 会议。计算。维斯, 2017 年, 第 5889–5898 页。[24] K. Kulkarni、S. Lohit、P. Turaga、R. Kerviche 和 A. Ashok, “ReconNet: 通过压缩感知测量对图像进行非迭代重建”, Proc. 中。IEEE 会议计算。维斯。模式识别, 2016, 第 449–458 页。[25] A. Mousavi、A. B. Patel 和 R. G. Baraniuk, “结构化信号恢复的深度学习方法”, Proc. 53 周年。阿勒顿会议交流。控制计算, 2015, 第 1336–1343 页。[26] J. 张和 B. Ghanem, “ISTA-Net: 用于图像压缩感知的可解释优化启发的深度网络”, Proc. IEEE Conf. 计算。维斯。模式识别, 2018, 第 1828–1837 页。[27] L. Boominathan、M. Maniparambil、H. Gupta、R. Baburajan 和 K. Mitra, “在不同测量量下进行傅里叶叠层描记术的相位检索”, Proc. 英国人。马赫。维斯。会议, 2018。[28] K. Monakhova、J. Yurtsever、G. Kuo、N. Antipa、K. Yanny 和 L. Waller, “学习了基于掩模的实用无透镜成像的重建”, 选项。快报, 卷。27, 没有。20, 第 28 075–28 090 页, 2019 年。[29] J.K. 亚当斯等人。 “带有超小型无透镜平面镜的单晶 3D 荧光显微镜”, Sci. 进展, 卷。3, 没有。2017 年 12 月, 艺术。不。e1701548。[30] S. J. Reeves, “无边界伪影的快速图像恢复”, IEEE Trans. 图像处理, 卷。14, 没有。10, 第 1448–1453 页, 2005 年 10 月。[31] O. Ronneberger、P. Fischer 和 T. Brox, “U-Net: 用于生物医学图像分割的卷积网络”, Proc. 中。国际。会议。医学图像计算。计算机辅助干预, 2015 年, 第 234–241 页。[32] J. Johnson、A. Alahi 和 L. Fei-Fei, “实时风格传输和超分辨率的感知损失”, Proc. 中。欧元。会议。计算。视觉, 2016, 第 694–711 页。[33] K. Simonyan 和 A. Zisserman, “用于大规模图像识别的非常深的卷积网络”, Proc. 中。国际。会议。学。习。Representations, 2015。[34] I. Goodfellow 等人, “生成对抗网络”, Proc. 国际会议神经信息。过程。系统, 2014, 第 2672–2680 页。[35] S. Gu、Y. Li、L. V. Gool 和 R. Timofte, “用于快速图像去噪的自引导网络”, Proc. 中。IEEE 国际。会议。计算。视觉, 2019 年, 第 17 页。2511–2520。[36] W. Shi 等人, “使用高效的子像素卷积神经网络实现实时单图像和视频超分辨率”, Proc. IEEE Conf. 计算。维斯。模式识别, 2016, 第 1874–1883 页。[37] P. Ramachandran、B. Zoph 和 Q. V. Le, “搜索激活函数”, Proc. 中。国际。会议。学习。陈述, 2018 年。[38] C. Ledig 等人, “使用生成对抗网络实现逼真的单图像超分辨率”, Proc. IEEE 会议计算。维斯。模式识别, 2017, 第 105–114 页。[39] R. Mechrez、I. Talmi 和 L. Zelnik-Manor, “使用非对齐数据进行图像转换的上下文损失”, Proc. 中。欧洲会议。计算。视觉, 2018, 第 768–783 页。[40] O. Russakovsky 等人, “ImageNet 大规模视觉识别挑战”, Int. J. 计算机。视觉, 卷。115, 没有。3, 第 211–252 页, 2015 年。[41] D. P. Kingma 和 J. Ba, “Adam: 一种随机优化方法”, Proc. 国际。会议。学习。陈述, 2015 年。[42] R. Zhang、P. Isola、A. A. Efros、E. Shechtman 和 O. Wang, “深度特征作为感知度量的不合理有效性”, Proc. IEEE 会议计算。维斯。模式识别, 2018 年, 第 11 页。586–595。



Salman Siddique Khan 于 2018 年获得印度鲁尔克拉国家理工学院电子与仪器工程学士学位。他目前正在印度泰米尔纳德邦金奈印度理工学院马德拉斯分校电气工程系攻读博士学位。他的研究兴趣包括计算成像、信号处理、光学和计算机视觉。



Varun Sundar 目前正在印度泰米尔纳德邦金奈印度理工学院马德拉斯分校电气工程系攻读本科学位。他也是威斯康星州麦迪逊市威斯康星大学麦迪逊分校即将获得博士学位。在印度理工学院马德拉斯分校，他在计算成像实验室工作，负责无透镜成像系统的研究。



Vivek Boominathan 于 2012 年在印度海得拉巴印度理工学院获得电气工程学士学位，并分别于 2016 年和 2019 年在德克萨斯州休斯顿莱斯大学电气与计算机工程系获得硕士和博士学位。他目前是德克萨斯州休斯顿莱斯大学的博士后。他的研究兴趣包括计算机视觉、信号处理、波动光学和计算成像领域。

Ashok Veeraraghavan 于 2002 年获得印度马德拉斯印度理工学院电气工程学士学位，并于马里兰大学帕克分校电气与计算机工程系获得硕士和博士学位，分别于 2004 年和 2008 年。他目前是德克萨斯州休斯顿莱斯大学的电气和计算机工程副教授。在加入莱斯大学之前，他在马萨诸塞州剑桥的三菱电机研究实验室担任了三年的研究科学家。他的研究兴趣包括计算成像、计算机视觉、机器学习和机器人领域。他的论文获得了马里兰大学电气与计算机工程系的博士论文奖。他是 2017 年美国国家科学基金会职业奖获得者。在莱斯大学，他负责计算成像和视觉实验室。

Kaushik Mitra 获得马里兰大学电气与计算机工程系博士学位。他目前是印度钦奈马德拉斯印度理工学院电气工程系的助理教授。在加入印度理工学院马德拉斯分校之前，他是德克萨斯州休斯顿莱斯大学电气与计算机工程系的博士后研究员。他的研究兴趣包括计算成像、计算机视觉和机器学习。他对计算成像的贡献包括提出用于分析和设计新型计算成像系统的理论框架、开发新型成像系统（例如混合光场相机和各种相机阵列）以及使用机器学习技术（例如字典学习和用于提高计算成像系统性能的深度学习）。

" 有关此或任何其他计算主题的更多信息，请访问我们的数字图书馆：  
[www.computer.org/csdn](http://www.computer.org/csdn)。