arXiv:2304.00306v2 [cs.CV] 2 Dec 2023

CapsFlow：使用胶囊网络进行光流估计

Rajat Arora*、Rahul Chand*、K Ram Prabhrakar 和 R Venkatesh Babu

印度科学研究所视频分析实验室

摘要

我们提出了一种框架，利用最近提出的胶囊网络来解决光流问题，这是计算机视觉领域的基础任务之一。现有的大多数先进深度架构要么使用相关性操作来匹配特征。尽管相关层对超参数的选择较为敏感，且不会对对象的底层结构施加先验，但时空特征会受到网络感受野的限制。此外，人类会将运动对象作为一个整体来观察，而这是相关性或时空特征无法编码的。另一方面，胶囊专门用于对独立实体及其姿态进行建模，并将姿态表示为连续矩阵。因此，我们表明，对胶囊检测到的对象姿态执行简单的线性操作，就足以对光流进行建模。我们在一个小型玩具数据集上展示了结果，性能优于 FlowNetC 和 PWC-Net 模型。

1 引言

光流估计是计算机视觉中的一项基础问题。给定一对图像，光流旨在求得稠密运动场，为每个像素分配一个向量位移，指示其在第二幅图像中移动到的位置。要估计两幅图像之间的位移场，需要学习精细的局部细节以及全局结构信息，以便在不同位置对图像进行匹配。

近年来，利用深度学习方法估计光流取得了巨大进展（Fischer 等人，2015年；Ilg 等人，2016；Meister 等人，2017年；Sun 等人，2017年）。尽管现有方法已取得良好效果，但大多数方法都依赖多阶段流程，并计算相关图或学习图像特征之间的时空特征，以辅助匹配过程。相关性通常通过计算给定邻域内、特定搜索空间中的一组像素之间的对应关系来实现。

形式化地说，给定两个特征图 $\mathbf{f_1}$,$\mathbf{f_2}$ with$w$, c, h being g its width, height and number of channels, 相关层会将$\mathbf{f_1}$ 中的每个图像块与$\mathbf{f_2}$ 中固定邻域内的每个图像块进行比较。图像块的大小（$\mathbf{K}$）等于$2k + 1$，其中 k 为卷积核大小。此外，为降低复杂度，并非所有图像块都会进行匹配。 f_1 中以 x_1 为中心的图像块，仅与 f_2 中中心 x_2 位于邻域内的图像块进行相关性计算；该邻域的大小为 $2d + 1$，中心位于x_1，其中 d 为位移参数。该操作的复杂度为 D^2cwh，输出大小为（$w \times h \times D^2$）。

因此，根据这些超参数的不同，相关性可能能够找到精确匹配，也可能无法找到。类似地，时空特征的学习会受到所使用滤波器感受野的限制。另一方面，胶囊天然适合这一任务，因为特定胶囊捕获的任何实体的位置和方向都由连续运动向量表示，而连续运动向量可以对各种类型的复杂刚体或非刚体运动进行建模。为此，我们在玩具数据集上展示了胶囊的有效性，并且据我们所知，首次将胶囊扩展到密集预测任务。

2 相关工作

胶囊网络：胶囊网络由（Sabour 等人，2017年）提出，是一种不同于卷积神经网络（卷积神经网络）的神经网络架构，其目标是实现视角等变性，而非不变性。胶囊网络通过使用一组称为胶囊的神经元来实现这一点，以相对于观察者编码实体是否存在及其姿态。

本稿作为本科毕业论文的一部分完成，并提交至 ICCV 2019 Neural Architects Workshop。已修正上一版本中会议名称的错误。

此外，他们使用动态路由机制，将姿态从一个胶囊层转换到下一个胶囊层。

尽管最初胶囊网络使用一个 8 维向量来编码实体的姿态，并以其范数表示实体存在的概率，但后来（Hinton 等人，2018年）将其修改为使用 4×4 矩阵表示实体的姿态，并用一个独立的标量表示实体是否存在。（Hinton 等人，2018年）还将（Sabour 等人，2017年）中的一致性路由替换为基于 EM 的路由。自问世以来，胶囊已用于二值分割（LaLonde 和 Bagci，2018年）、动作识别（Duarte 等人，2018年）、三维点重建（Zhao 等人，2018年）以及许多其他应用。

光流方法：经典方法通常通过能量最小化来估计光流，尤其是在（Horn 和 Schunck，1981年）的工作之后。然而，能量最小化方法无法处理大位移（（Brox 等人，2004年）），因此一系列研究（（Barnes 等人，2010年；Liu 等人，2011；Xu 等人，2012））采用描述子或特征匹配，并结合由粗到细进行的能量最小化，以缓解这一问题。（Weinzaepfel 等人，2013）将上述方法与深度学习相结合，在能量最小化之前使用人工选择的卷积滤波器提取多个尺度的特征

卷积神经网络已广泛应用于高层视觉问题，光流计算也不例外。（Fischer 等人，2015年）提出了 FlowNetC，该网络利用一组学习到的特征之间的相关性，并以多尺度方式通过卷积神经网络层对其进行细化。SpyNet（Ranjan 和 Black，2016）将两幅输入图像堆叠在一起，以在多个尺度上学习时空滤波器，并学习残差流，通过层次化细化来改进输出。PWC-Net（Sun 等人，2017年）将两者结合起来，利用学习到的特征之间的相关性，在多个尺度上估计视差，随后通过层次化细化生成最终光流。此外，由于获取用于光流的真实世界数据是一项艰巨任务，（Meister 等人，2017年；Wang 等人，2017年；Zou 等人，2018年）探索了该任务的无监督方法，并以 FlowNetC 架构为基础，采用基于双向变形的损失以及额外的平滑性约束，以获得令人满意的性能。

我们的工作偏离了这一基本范式：我们试图利用胶囊网络更强的表征能力，借助其中的运动向量，隐式地建模物体可能经历的不同类型的变换。

3 总体方法

直觉： 卷积神经网络中的单个神经元能够捕捉图像中某个特征的存在或缺失；而单个胶囊不仅能捕捉某个实体是否存在，还会输出一个 4×4姿态矩阵，从而学习该实体相对于观察者的空间关系（平移、旋转等）。我们利用这一特性，将实体（物体）的运动计算为其姿态矩阵之间的变换。这就将计算开销巨大的相关层简化为单个操作。重新表述这一问题的直觉在于，胶囊的姿态矩阵能够学习其所描述实体的位置参数，因此，通过使用适当的变换（例如，对两幅图像的姿态矩阵做差），便可以捕捉实体在两幅图像之间的运动

胶囊更适合分类任务，因为每个类别胶囊都试图对一个独立实体建模。但是，真实场景以及传统光流数据集通常都没有预先设定的类别列表。因此，我们通过一个玩具数据集展示胶囊网络在类别监督情况下的有效性。该数据集包含五种不同形状（或类别），其大小和方向各异，如图3所示。

3.1 胶囊网络

在解释 CapFlow架构 之前，我们先简要介绍标准胶囊网络的主要组成部分：

卷积编码器： 胶囊网络使用少量卷积层和 ReLU 激活函数，从输入中提取基本特征。随后，这些特征将作为主胶囊层的输入。

胶囊层： 卷积层的输出分别传递给一组胶囊，这些胶囊执行卷积，以输出一个 4×4变换矩阵 M，表示检测到的实体的姿态，以及一个激活概率 A。随后，初级胶囊的输出通过

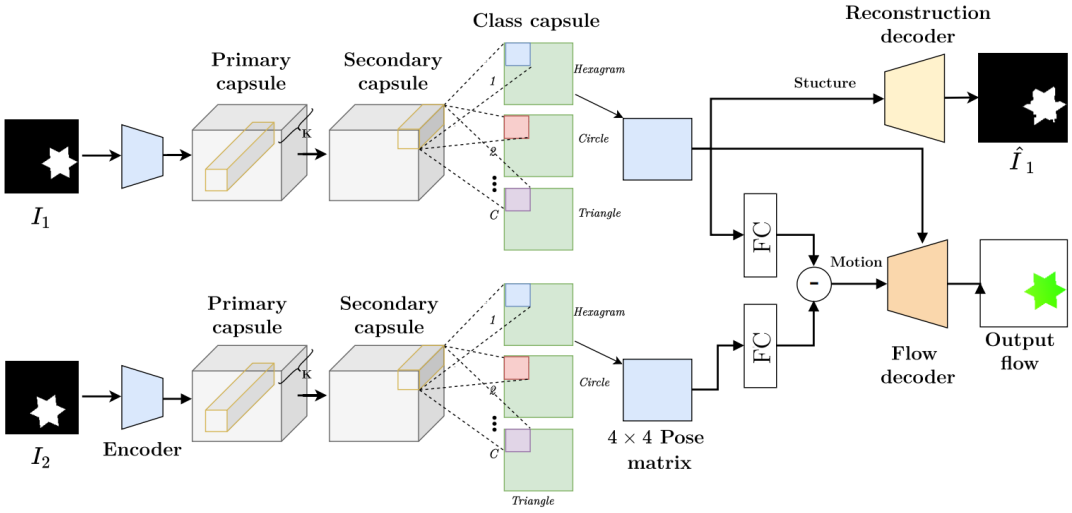

图1：该架构的第一部分由一个孪生网络组成，一对输入图像——图像1（I_1）和图像2（I_2）——经过卷积编码器处理。随后经过3层胶囊（包括类别胶囊），为每个 C （=5）个可能的类别输出一个 4×4 姿态矩阵。我们选择属于两幅图像平均激活值最高的类别的矩阵。然后，将两个编码传入全连接层，以形成更丰富的嵌入。接着，我们对这两个嵌入进行相减，得到运动嵌入，再将其传入卷积解码器，以获得输出流场图。第一幅图像的姿态矩阵还通过跳跃连接传入光流解码器，以细化光流。这些姿态矩阵也用于重建输入图像 I_1，其充当正则化器。

各相邻层胶囊对之间的可训练变换矩阵 W 路由至次级胶囊。第 L 层中胶囊 i 的姿态矩阵与 W_{ij} 相乘，得到 $V_{ij} = M_i W_{ij}$，它表示该胶囊对第 $L+1$ 层中 j^{th} 胶囊的投票。随后，第 $L+1$ 层中的姿态使用非线性路由机制（下文将作说明）来确定其姿态和激活向量。最终胶囊层由一个代表最终分类标签的单一胶囊组成。

胶囊之间的路由： 4×4 父胶囊姿态矩阵中的每个元素都属于一个高斯分布，其参数通过期望最大化（EM）算法进行估计。总体而言，期望步骤旨在计算第 L 层中的每个胶囊由第$L+1$层中的胶囊解释的概率。随后，最大化步骤尝试根据来自第 L层的输入，最大化第 $L+1$层中给定胶囊的激活值。

3.2 CapsFlow网络架构

Siamese CapsFlow架构概览如图1所示。网络的输入是一对 128×128×3 RGB图像，即I1和I2。网络首先经过四个卷积层，卷积核大小为 3×3 ，步幅交替设置为

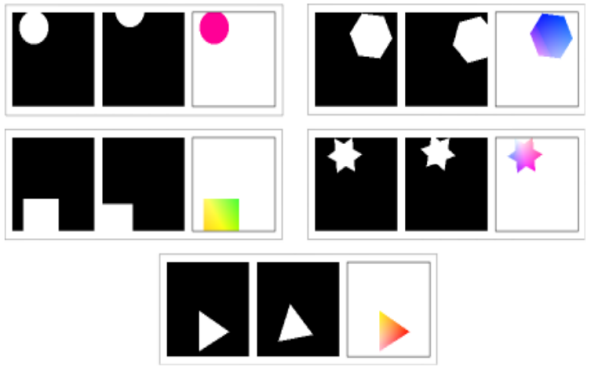

图2：用于训练和验证的输入图像及真实流场示例。

2和1，每层均采用ReLU非线性激活函数。所得维度为**32×32×256** 的特征图通过步幅为2的 3×3 卷积操作转换为维度为 **16×16×16×17**的胶囊层（**K×H×W×4*4+1**，其中 K=表示胶囊类型的数量）。随后是第二个胶囊层，该层包含16种胶囊类型，具有 7×7 感受野，步幅为2。

随后，第二个胶囊层连接到最终的卷积胶囊层（类别胶囊），该层包含 C 种胶囊类型，其中 C 是数据集中形状的数量。由此得到的输出

（**C×6×6×（16+1）**）由两个部分组成：姿态矩阵（**C×6×6×16**）和激活向量（**C×6×6×1**）。请注意，我们不通过坐标相加来计算最终胶囊层的姿态和激活值，因为倒数第二个胶囊层与最终胶囊层并非全连接，因此不会丢失空间信息（（Hinton 等人，2018年；Duarte 等人，2018年））。从二维胶囊层（**C x H x Wx17**）路由到一维胶囊（**C×17**）时，需要进行坐标相加以保留空间信息。每种形状的激活概率通过对该形状激活向量的空间维度（**6×6**）取平均值计算。激活概率最高的胶囊对应于网络预测的形状。

随后对类别胶囊层（C×6×6×16）的输出姿态进行掩蔽，以获得与真实值形状相对应的姿态矩阵。在训练期间，我们利用真实值信息选择正确的类别；而在推理期间，则根据均值激活向量最高的类别进行选择。掩蔽后的输出（6×6×16）会传入一个带有 ReLU 非线性的全连接层，以获得形状的结构嵌入（8×8×16）。$I1$和 $I2$ 的结构嵌入均传入共享卷积层，随后将二者相减以计算运动嵌入，该嵌入编码了从 I_1 到 I_2的运动。

为了帮助网络学习正确的姿态参数，我们将结构嵌入传入一个由四层组成的转置卷积解码器，以重建输入图像 I_1。这是必要的，因为不使用重建解码器进行训练会导致网络学习到较差的结构嵌入，进而影响运动嵌入。为了构建最终的光流图，我们将运动嵌入和结构嵌入传入一个由四个转置卷积层组成的解码器，每层的步幅为 2，卷积核大小为4×4。对解码器的输出（64×64×2）进行外推，以获得形状为 128×128×2的预测光流图。

3.3 目标函数

目标函数由三种损失组成：a）均方重建损失，用于学习正确的姿态嵌入；b）激活损失，用于学习正确的目标类别；c）真实光流图与预测光流图之间的端点误差（EPE）损失。为了计算重建

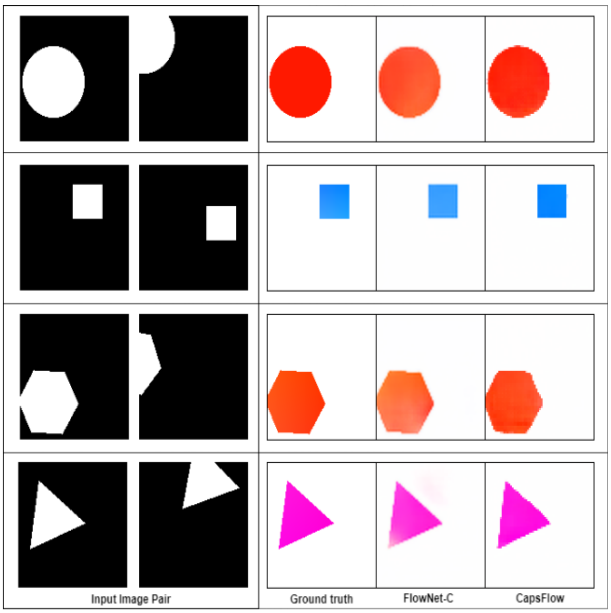

图3：少数示例表明，FlowNetC 低估了光流，而我们提出的模型输出结果更接近真实值。（彩色放大查看效果最佳。）

损失，我们计算 $I_1\hat{}$ 与重建图像 I_1之间的均方误差。

$$L_{mse} = ||I_1 - \hat{I}_1||^2$$

通过实验我们发现，重建损失起到正则化器的作用，有助于确保姿态矩阵学习到所有相关特征。在没有重建损失的情况下，网络无法收敛，可能是因为缺乏足够的监督来学习正确的姿态矩阵（光流仅取决于它们的差值，而不取决于实际值）。我们使用间隔损失来训练激活向量，其计算方式如下：

$$L_m = \sum_{i \neq t} max[0, m - (mean(a_t) - mean(a_i))]^2$$

其中，a_t 是目标形状类别的激活值，ai 是对应于胶囊 i的激活值。在计算损失之前，我们取激活值的均值。在测试阶段，使用激活值最高的胶囊来确定用于相减以估计光流的姿态。训练期间，间隔 m 设为 0.95。EPE损失的计算方式如下：

$$L_{epe} = ||flow_{gt} - flow_{pred}||^2$$

因此，CapsFlow网络使用以下目标函数进行训练

$$L = L_{epe} + \lambda_1 L_{margin} + \lambda_2 L_{mse}$$

编码器					
卷积步幅	3x3	3x3	3x3	3x3	
滤波器数量	2	1	2	1	
批归一化	Yes	Yes	Yes	Yes	
解码器					
转置卷积	4x4	4x4	4x4	-	-
卷积	-	-	-	1x1	1x1
步幅	2	2	2	1	1
滤波器数量	64	64	32	16	2
重建解码器					
转置卷积	4x4	4x4	4x4	4x4	-
卷积	-	-	-	-	1x1
步幅	2	2	2	1	1
滤波器数量	32	64	16	8	1

表 1：CapsFlow网络架构

4 实验

为检验我们方法的有效性，我们准备了一个由简单形状组成的玩具数据集，如图3所示，并将我们的方法与 FlowNetC（Fischer 等人，2015年）以及 PWC-Net（Sun 等人，2017年）进行比较。我们还通过在 Flying Chairs 数据集上进行实验，探索将我们的方法扩展到真实世界图像的途径

4.1 训练

通过随机采样一种形状类型（正方形、六边形、三角形、圆形和六边形），以及 0 到 128 之间的随机 (x, y) 位置来生成数据。随后对该形状进行随机旋转和平移，以获得相应的第二张图像。数据在线生成，以防止过拟合。我们固定使用一组包含 2500 张图像的数据进行测试。表 1 给出了所使用的编码器和解码器的架构详情。

重建解码器和光流解码器可以同时训练，也可以按顺序训练，即先训练重建解码器，再训练光流解码器。式4中的 λ_1 和 λ_2 的值分别设为0.05和2.5。所有模型均在NVIDIA GTX 1080Ti GPU上进行训练。CapsFlow总共训练了30K次迭代，迷你批次大小为64。对于顺序训练情况，首先对重建解码器进行10k次迭代的预训练。路由过程中的EM 迭代次数设为 3，与（Hinton 等人，2018年）中的设置相同。我们使用Adam优化器（Kingma 和 Ba，2014），并将学习率固定为0.001。

模型	EPE	参数量（百万）	时间（毫秒）
CapsFlow（T）	0.48	1.62	13.39
CapsFlow（S）	0.39	1.62	13.39
FlowNetC	0.50	39.17	3.1
PWC-Net	0.47	8.75	12.1

表2：单形状数据集上的定量比较：T和S分别表示同时训练和顺序训练，更多信息请参阅第4.1节。平均时间根据分辨率为128x128的输入对计算得出。最佳结果以红色标出，次佳结果以蓝色标出。

4.2 结果

我们将结果与 FlowNetC 和 PWC-Net 进行了比较。这两个模型均使用端点误差进行训练，迭代 90k 次，批量大小为 16，并采用学习率为 0.001 的 Adam优化器，与所提出的 CapsFlow 类似。表 2 展示了两种 CapsFlow 模型的端点误差比较。两种 CapsFlow 变体的表现均优于 FlowNetC 和 PWC-Net，同时还拥有 8× 更少的参数。不过，尽管参数数量较少，胶囊的耗时仍显著更长，这主要是因为迭代式EM路由过程。但随着胶囊网络不断发展并得到更广泛的应用，硬件加速和更完善的路由过程很有可能降低其时间开销。

在可视化 CapsFlow 和 FlownetC 的光流输出时，我们注意到，后者如图3所示，在大量样本中低估了光流幅值。为确认这一点，我们在另一个验证集上测试了这两个模型，该验证集的平均光流幅值是训练集的 1.5 倍。FlownetC 在这个新数据集上的端点误差从 0.50 增至 1.51，增加了 3 倍，而 CapsFlow 的端点误差仅从 0.39 增至 0.81。这凸显了 CapsFlow 更强的泛化能力。我们推测，产生这一现象的原因是：胶囊将光流建模为连续运动矩阵，而相关性对位移和邻域大小的选择较为敏感，因此可能无法同样良好地建模大幅值光流，尤其是在训练过程中未见过这些光流的情况下。我们还尝试观察姿态矩阵和差值矩阵通过在 [-1,1] in 范围内以 0.5 为增量统一缩放其幅值时所建模的内容。在改变差值的幅值时，我们看到光流

从光流谱的一端极值变为完全相反的另一端，但整体结构没有变化（图4）。另一方面，将跳跃连接的幅值在 [0.25, 2.5]in 范围内以 0.25 为增量进行改变，会改变形状尺度，如图5所示。这表明，光流和姿态本质上实现了解耦：光流来自姿态差异，而结构来自图像 1的姿态矩阵。

扩展到多种形状：我们通过在每张图像中动态采样两种不同的形状，生成 MultiShape 数据集的实例。将两种形状的光流合并，得到真实光流图。平均而言，每种形状的边界框大小为 50×50，两种形状的中心均位于大小为 90×90 的区域内。因此，两个边界框平均有 45% 的重叠。

在使用多种形状训练 CapsFlow 时，我们每次屏蔽一个类别胶囊，并使用其姿态矩阵构建相应的光流图。网络的最终输出通过叠加两张光流图得到（在两张图相交的区域，我们按照创建训练集时采用的相同优先级，仅保留一种形状的光流）。本实验的训练参数和损失与单一形状实验相同。与 FlowNetC 的 1.9 相比，我们取得了略好的端点误差 1.78。对于测试图像，我们选取激活程度最高的两个类别胶囊，并使用它们的姿态矩阵分别构建光流图（之后用这些光流图生成最终光流图）。CapsFlow 的一个优势是，与其他方法不同，它能够分别获得图像中每种形状的光流图。由于胶囊对每个物体的形状施加了强约束，即使某种形状受到严重遮挡，CapsFlow 也能利用其对物体形状的先验知识推断出完整物体，如图 6 所示。

4.3 无监督训练

我们还尝试测试所提出模型在无监督设置下的有效性。为了在无监督设置下训练 CapsFlow，我们对最终层中所有胶囊的输出取平均，而不是对它们进行掩蔽。我们使用 UnFlow [(Meister 等人，2017年)], 所采用的同一组损失来训练 CapsFlow：a）输入图像与其光流扭曲对应图像之间的遮挡感知光度损失；b）二阶

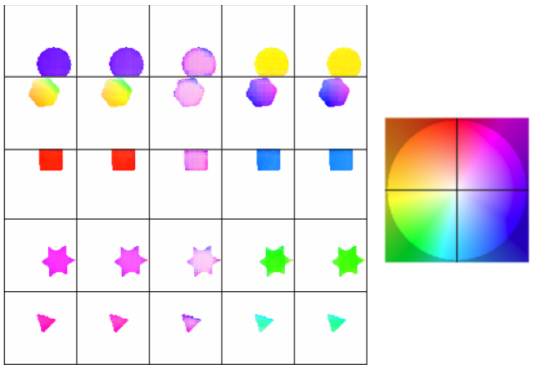

图 4：左：光流插值结果。我们在保持给定图像的姿态之间的差值矩阵与 I1 的跳跃连接不变的情况下，缩放差值矩阵的幅度，并注意到差值矩阵只会改变光流，而不会改变结构。具体而言，缩放幅度似乎等价于从光流谱的一端移动到 diametrically opposite 的另一端。右：光流颜色编码。

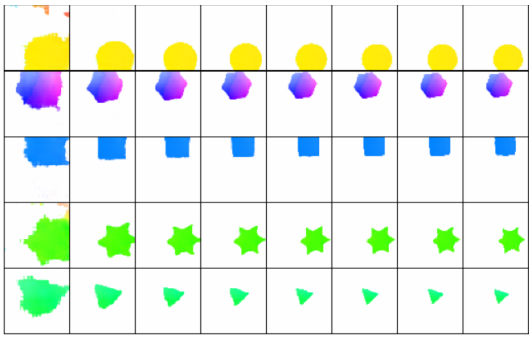

图 5：我们改变姿态矩阵的幅度，从 I_1 开始，同时不改变差值，并注意到只有光流的输出形状发生变化，而光流值本身没有任何改变。

平滑度约束，用于鼓励相邻光流保持共线性；以及 c）前向-后向一致性惩罚。在 Shapes 数据集上，我们的模型表现与 UnFlow 方法相当，端点误差为 2.05，而 UnFlow 方法的端点误差为 1.8。不过，尽管端点误差得分不错，我们注意到在没有掩蔽的情况下，5 个类别中始终只有 1 个处于激活状态；即使不考虑其他胶囊，也能取得相当的结果。下一节将阐述其中的原因

5 缺点与未来工作

尽管我们提出的模型在玩具数据集上取得了令人鼓舞的结果，但胶囊网络仍无法直接用于真实世界样本。当潜在胶囊不代表不同类别时，胶囊往往在相关任务中表现不佳 [13]。性能欠佳是由于胶囊学习

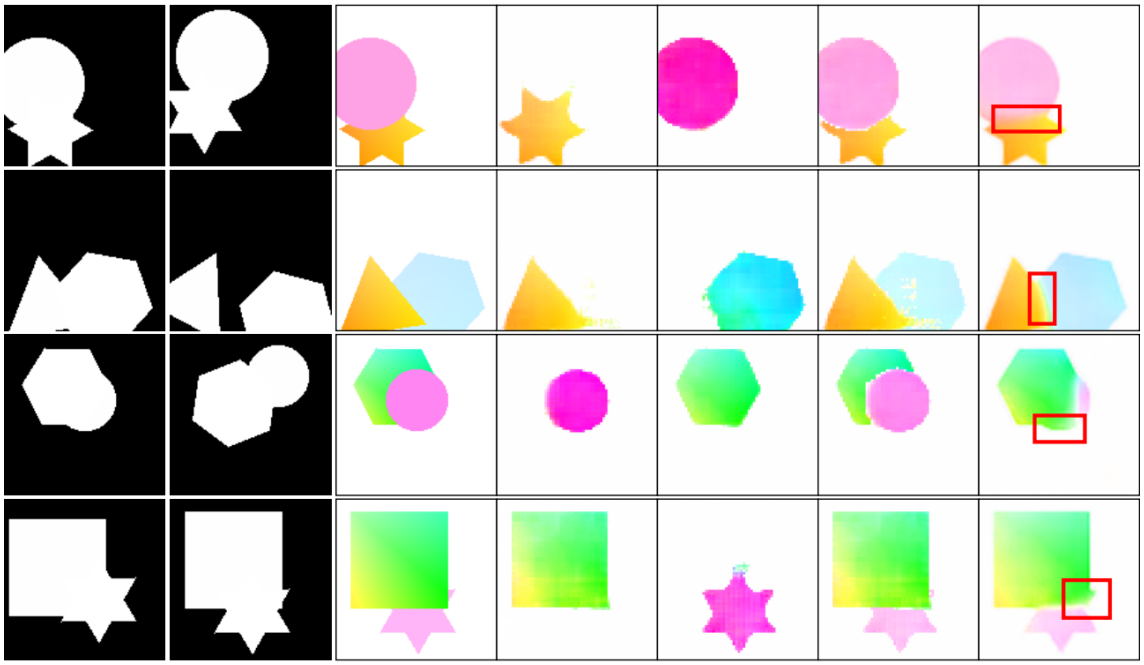

图 6：即使在光流高度重叠的情况下，CapsFlow 也能成功建模各个光流。此外，在运动边界和重叠区域，由于缺乏关于形状边界的先验信息，FlowNetC 会如预期般将属于一种形状的光流分配给另一种形状（红框标出）

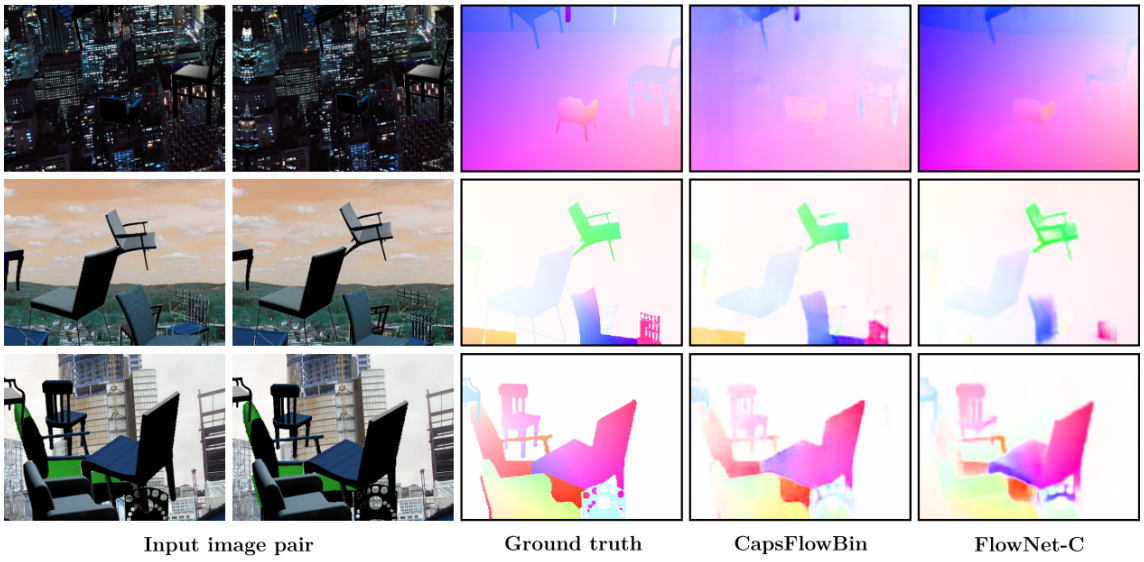

图 7：FlyingChairs 数据集上的定性结果。我们提出的 CapsFlowBin（采用分箱的 CapsFlow）方法取得了 3.6 的端点误差，而 FlowNetC 的端点误差为 2.0。

以不同范式学习参数。胶囊并非通过 EM 路由（生成式方式）学习所有参数，而是通过反向传播以判别式方式学习变换矩阵；如果缺少这一过程，可能会得到退化解，例如所有变换矩阵都坍缩为零。

如果没有具体的类别胶囊，变换矩阵就没有动力或梯度彼此产生差异。在这种情况下，某个低层胶囊子集被变换到两个高层胶囊姿态空间中的相似簇的概率 j 和 k 会增加。这会导致部分到整体的关系崩溃，因为二者 j 和 k 现在对非常相似的实体进行建模，进而导致路由到 j 和 k 的低层胶囊也对相似实体进行建模 [（Rawlinson 等人，2018年）]。因此，未进行掩码的潜在胶囊网络所需的胶囊数量是其进行掩码的对应网络的指数倍。这是一个重大缺点，因为这种要求具有很强的限制性：大多数密集预测任务都没有（明确或显式的）类别/实体，例如深度估计、超分辨率和三维分割。胶囊网络的另一个缺点是，除非经过显式训练，否则每个类别胶囊只能对该实体的单个实例进行建模，因此在场景中存在同一实体的多个实例时会失效。此外，胶囊也无法对全局运动进行建模，不过这并不是一个主要问题，因为已有其他稳健技术可用于估计相机运动。

我们尝试过一种稠密光流的变通方法：将真值在空间上分箱，并将这些分箱视为类别胶囊。这些类别胶囊的输入通过从次级胶囊层本身中减去姿态得到。该方法在训练期间效果良好，但在测试时因误分类率过高而失效（见图 7）。

尽管存在这些缺点，胶囊似乎仍是一个有前景的研究方向，并能解决当前基于卷积神经网络的深度方法在跟踪和光流计算中面临的不足。

6 结论

根据上述实验，我们得出结论：胶囊网络尽管存在各种不足且仍处于起步阶段，但为运动建模和光流计算提供了一条很有前景的途径，尽管在玩具数据集上，无论是单个形状还是多个形状的情形，它们的表现都优于当前最先进的方法，并且展现出更强的泛化能力，但要使其适用于实体清单不完整或场景中存在同一实体的多个实例的情况，仍需开展更多研究。尽管如此，我们相信，胶囊网络凭借更强的表征能力以及对物体间空间关系的隐式编码，是运动估计和光流领域值得进一步研究的方向。我们认为，随着胶囊网络不断改进，我们的框架可以适用于真实世界图像，从而获得更好的结果。

参考文献

Connelly Barnes、Eli Shechtman、Dan B. Goldman 和 Adam Finkelstein。2010年。广义 PatchMatch 对应算法。发表于欧洲计算机视觉会议。

Thomas Brox、Andrés Bruhn、Nils Papenberg 和 Joachim Weickert。2004年。基于变形理论的高精度光流估计。发表于欧洲计算机视觉会议。

Kevin Duarte、Yogesh Singh Rawat 和 Mubarak Shah。2018年。Videocapsulenet：一种用于动作检测的简化网络。载于神经信息处理系统。

Philipp Fischer、Alexey Dosovitskiy、Eddy Ilg、Philip Häusser、Caner Hazirbas、Vladimir Golkov、Patrick van der Smagt、Daniel Cremers 和Thomas Brox。2015年。FlowNet：使用卷积网络学习光流。*CoRR*，abs/1504.06852。

Geoffrey E. Hinton、Sara Sabour 和 Nicholas Frosst。2018年。带 EM 路由的矩阵胶囊。载于国际学习表征会议。

Berthold K.P. Horn 和Brian G. Schunck。1981年。确定光流。人工智能，17(1)：185–203。

Eddy Ilg、Nikolaus Mayer、TonmoySaikia、Margret Keuper、Alexey Dosovitskiy 和 Thomas

Brox。2016。Flownet 2.0：使用深度网络进行光流估计的演进。*CoRR*，abs/1612.01925。

Diederik P. Kingma 和 Jimmy Ba。2014。Adam：一种随机优化方法。*CoRR*，abs/1412.6980。

Rodney LaLonde 和 Ulas Bagci。2018年。用于目标分割的胶囊。*ArXiv*，abs/1804.04241。

Ce Liu、Jenny Yuen 和 Antonio Torralba。2011 年。Sift flow：跨场景稠密对应及其应用。*IEEE* 模式分析与机器智能汇刊，33(5):978–994。Simon Meister、Junhwa Hur 和 Stefan Roth。2017 年。Unflow：利用双向 Census 损失进行光流无监督学习。载于*AAAI* 人工智能会议。Anurag Ranjan 和 Michael J. Black。2016 年。使用空间金字塔网络进行光流估计。*2017 IEEE* 计算机视觉与模式识别会议 *(CVPR)*，第 2720–2729 页。David Rawlinson、Abdelrahman Ahmed 和 Gideon Kowadlo。2018 年。稀疏无监督胶囊具有更强的泛化能力。*ArXiv*，abs/1804.06094。Sara Sabour、Nicholas Frosst 和 Geoffrey E. Hinton。2017 年。胶囊之间的动态路由。*ArXiv*，abs/1710.09829。Deqing Sun、Xiaodong Yang、Ming-Yu Liu 和 Jan Kautz。2017 年。PWC-Net：利用金字塔、变形和代价体进行光流估计的卷积神经网络。*2018 IEEE/CVF* 计算机视觉与模式识别会议，第 8934–8943 页。Yang Wang、Yezhou Yang、Zhenheng Yang、Liang Zhao 和 Wei Xu。2017 年。遮挡感知的光流无监督学习。*2018 IEEE/CVF* 计算机视觉与模式识别会议，第 4884–4893 页。Philippe Weinzaepfel、Jérôme Revaud、Zaïd Harchaoui 和 Cordelia Schmid。2013 年。Deep-flow：基于深度匹配的大位移光流。*2013 IEEE* 国际计算机视觉会议，第 1385–1392 页。

Li Xu、Jiaya Jia 和 Yasuyuki Matsushita。2012 年。保留运动细节的光流估计。*IEEE* 模式分析与机器智能汇刊，34(9):1744–1757。Yongheng Zhao、Tolga Birdal、Haowen Deng 和 Federico Tombari。2018 年。3D 点胶囊网络。*CoRR*，abs/1812.10775。Yuliang Zou、Zelun Luo 和 Jia-Bin Huang。2018 年。Df-net：利用跨任务一致性进行深度与光流的无监督联合学习。*CoRR*，abs/1809.01649。