A. 模型训练

原理：利用迁移学习（卷积和池化层学习到的层级特征是general的，因而可以被不同模型共享）

过程：预训练一些用于不同视频工作的CNN，再用新的分类器替换原来的分类器，最后按照三种策略对新模型进行微调：重新训练余下所有层，只训练新分类器，以及二者的折中。新任务的数据集越少，推荐重新训练的层数也越少（迁移学习的常识，防止过拟合）

这里还有一个Tradeoff问题：新旧模型和新模型之间共享的层数越多，迁移训练速度就能越快，且在线推理时计算开销就会越小，但相应的每个新模型各自的特征就不能充分学习，会变得less specific，导致精度的下降。因此合理选择冻结层的数量来平衡计算开销和推理精度很重要。这里作者选择用第三个策略来进行训练，但没有解释retrained和frozen层的比例选择，只是一笔带过。

B. 模型分割

把模型分成特征提取模型和特征推理模型，放在一个移动设备分布式集群上的不同设备上运行。这里模型的分割点有多种选择，理论上说分割点选的层数越高，传输的特征大小就越小。这是因为从较低层提取的特征通常代表着较低层次的信息，细节会更多，且冗余信息也很多。但这不代表我们应该一味地选择更高的分割点，这是因为和传统的不对称的端设备到云服务器的架构不同，云服务器上计算资源充沛，其计算开销可以忽略不计，通信开销占主导；而对于本系统的移动设备分布式架构，设备资源紧张，计算开销和通讯开销同样重要。选择更高的分割点可以降低通信开销，但这会导致两个设备的计算开销严重不平衡，一个开销少，一个开销多，很明显不能充分利用本就不多的资源，从而拉低吞吐率，还会反而加重延迟。那么如何选择分割点位置呢？本文也没有交代这个细节，只是说会根据冻结层的情况选择合适的分割点，因为冻结层需要共享，肯定不能被分割掉，除此之外就一笔带过了。此外，本文在评估环节还提到，他们设计了一个根据用户需求和环境因素自适应选取分割点的系统，但我并没有找到他们关于这个自适应算法的任何介绍，也只是一笔带过。

C. 特征压缩

AMVP采用了基于量化的压缩方式，来降低通讯开销。也就是把原先浮点类型的特征Tensor按照这个公式来转换成整数类型，从而大大减少了需要传输的特征大小。而在FI处，会根据这个公式的逆公式来还原得到原先的特征张量。由于存在取整等近似，还原前后的张量必然不会完全相等，但根据实验，当量化精度n达到一定阈值后，这个误差是可以忽略不计的。作者实验得到n=8，即将特征张量量化成8位整数效果比较好。而在基于量化的压缩基础上，还可以采用传统的压缩方式，如gzip,zlib等，进一步压缩需要传输的特征。（future work）

特征压缩的评价方法：压缩率（CR），保真度（Fidelity），压缩效益（CPB）

One-hot 独热编码

信息论的东西，学数电的时候好像做过这样的题，就是说用n位寄存器来表示n个状态，这样可以将离散的分类特征值映射成二进制向量，进而可以用汉明距离表示特征间的相似性。比如食堂有韵酒、东园、东教工，若用3位表示3个不同的食堂，则可以分别表示成001,010,100。

Hamming distance 汉明距离

一年前刷力扣的时候见过，就是说两个整数值对应位的差距。如0100和1101，其汉明距离就是2。

OTR：以原来的形式传输特征需要的延迟

QT：量化特征需要的延迟

TR：传输量化压缩后的特征需要的延迟

DQ：解量化得到原来特征需要的延迟

D.模型选择和任务分配

记号：

S：视频分析任务的集合

As, Ls, Ts：用户设置的对任务s的精确度、延迟和吞吐量的期望要求

Ms：对于任务s，可用的CNN模型集合

ms：拥有特定的预训练网络类型和冻层数量的一个竞选模型

加粗ms：一对模型的分割

f(m1s,m2s)：从m1s到m2s传输特征的大小

b(k1,k2)：从设备k1到设备k2的带宽

u(k,m)：设备k上分配给模型m的计算资源占总资源的比例

l(k,m)：设备k上分配100%资源给模型m时的处理延迟

D：传输延迟

.tflite格式的模型集和对应的profile文件都已经提前部署在移动设备上了，所以AMVP剩下的工作就是为每个任务选取合适的模型，并将整个pipeline合适地划分给集群中的设备完成。

要最小化损失函数，就要求第一项促使选择精度接近目标甚至超过目标的模型；第二项和第三项分别对选择了高于目标延迟和低于目标吞吐量的模型的惩罚。又因为本文中的视频流是固定帧率传输的，对于低于目标延迟和高于目标吞吐量的模型不会有额外收益。（感觉这里可以引入之前读的一些更改旋钮配置的思想来优化这个损失函数？）

对于这个损失函数，AMVP采取的解决策略是最小化拥有最大损失的任务的损失，即minMaxCost，从而归结为一个最优问题，并通过贪心算法解决。

u：分配给模型的计算资源；U：设备上可用的总计算资源

r：模型占用的运行时内存；R：设备上可用的总内存

TaskAssign()采用的是遗传算法，但没有对如何将assignment编码为一个二进制串作解释说明。

E.实验评估

移动设备之间通过wifi连接，Manpack是啥？？？没有说Manpack和移动设备之间的连接方式。

Fig 9先上升后下降：训练集少。冻结层少，导致过拟合；冻结层多，独特的特征难以提取。

F.值得讨论的地方：

对现有方法的整合，没有novelty。

迁移学习、模型层共享、模型量化、贪心算法、遗传算法等都是已有的且常用的工作。

唯一比较新的点在于模型分割与移动分布式集群的结合

迁移学习的问题

目的是完成多任务检测，并且希望通过迁移学习共享神经网络层来降低开销，那么为什么不直接使用多任务学习呢？多任务学习得到的网络不就是用一张网络来完成所有任务吗，这样所有任务的网络层本身就是相同的了。作者给出选择迁移学习的理由之一是可以微调冻结层数来达到精细控制开销与精度的权衡，但这样也未必更优于多任务学习的效果吧？还有一种可能是作者用来训练新模型的数据集不够大（500张图片做训练集，250张做验证集），这时采用迁移学习是合理的，但如果能取得足够大的数据集呢？这时迁移学习需要重新训练更多的层，是不是多任务学习就会更好一点？

对移动手机搭建分布式集群的合理性：

有没有实际意义，还是只是为了做实验而搭建。因为这个基于移动设备的分布式集群，设备之间通过同一个Wifi网络相连。这就要求在使用这个系统提供的视频分析服务时，首先每个设备都得必须安装一个客户端（内置model），且必须有一个唯一的服务端（包含profile的数据库和执行模型选择、任务调度）；wifi连接和必须安装app都是比较大的限制；而且在手机资源比较珍贵的背景下，你还得容忍自己的手机资源被别人拿去用来跑别人的任务。我不认为会有用户希望这么做。

并且和在单设备上运行多网络这个基线相比，实验得到AMVP降低了接近1s的延迟，但这是在保证网络带宽非常好的实验条件下进行的。但在比较不同带宽时，实验并没有比较AMVP和单设备基线，只比较了他们和另一个基线，很明显是在回避他们系统设计的网络问题，不能让人信服。

网络通信：问题很大！！！体现在特征量化的测试图和带宽比较的侧视图上。

想了解学长们和老师对遗传算法解最优解问题的看法（就像自然选择一样，随机选择初始父辈，随机突变和重组，感觉随机的成分有点太大了，不知道其效率究竟怎么样）

整篇论文出现了5处语法错误，1处标记错误