队伍介绍

队名：CASIA-AIRIA。

队员：史磊（博士在读），程科（博士在读）。

指导教师：张一帆副研究员。

单位：中国科学院自动化研究所，中国科学院自动化研究所南京人工智能芯片创新研究院。

竞赛介绍 [1]

今年 5 月，美图公司联合中国模式识别与计算机视觉学术会议（PRCV2018）共同举办的 PRCV2018「美图短视频实时分类挑战赛」正式开赛。来自中科院自动化所、中科院自动化所南京人工智能芯片创研院的史磊、程科在张一帆副研究员的指导下获得了 PRCV2018「美图短视频实时分类挑战赛」冠军。不同于以往只关注分类精度的比赛，本竞赛综合考察「算法准确率」和「实时分类」两个方面，将运行时间作为重要指标参与评估，将促进视频分类算法在工业界的应用。以下是冠军团队对本次挑战赛的技术分享总结：

- **数据集介绍**

本次竞赛使用的短视频数据集（MTSVRC 数据集）一共有 100,000 个视频，其中训练集有 50,000 个视频，验证集和测试集分别有 25,000 个视频。视频主要以短视频为主，长度约为 5 - 15s。数据集包含 50 个分类，视频类别包括舞蹈、唱歌、手工、健身等热门短视频类型，除了包含与人相关的一些行为类别，还有一些风景，宠物等类别。图片 1 展示了一些数据样例：

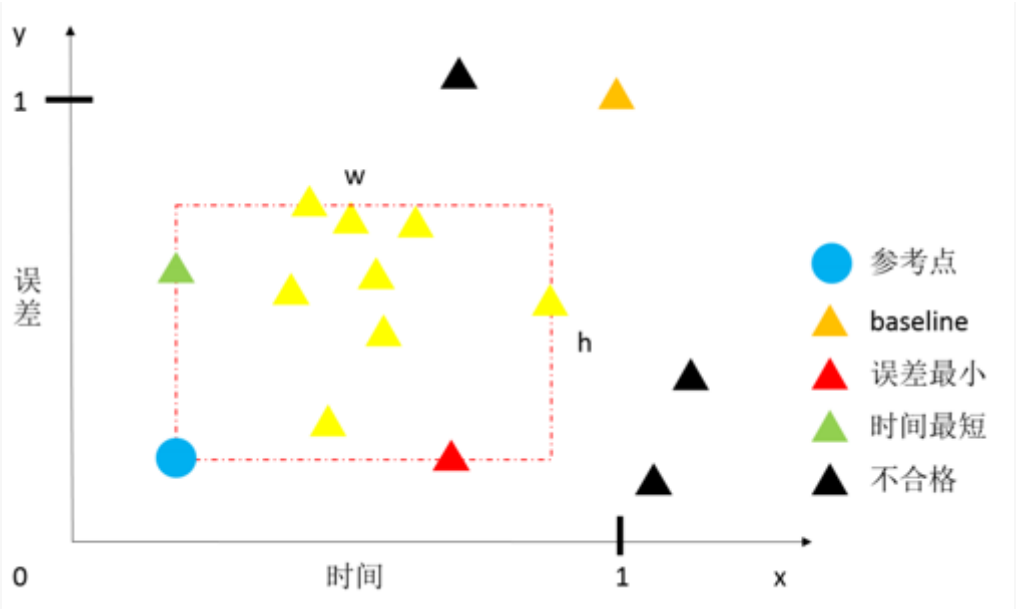
图片 1 数据样例

由于这些数据的主要来源为手机拍摄的日常视频，视频的大小，形状以及拍摄条件（例如光照，景深）等都不统一，造成了很大的类间差异与类内差异。同时，由于后期处理，视频经常会有一些特效和与类别无关的文字，也增加了视频识别的难度。图片 2 展示了一些困难样例，这些样例对模型的设计带来了很大的挑战。

图片 2 困难样例

- **评测方法**

由于竞赛同时考虑时间和精度，所以以往的分类误差不足以评测模型性能。图片 3 展示了此次竞赛所用的评测方法。



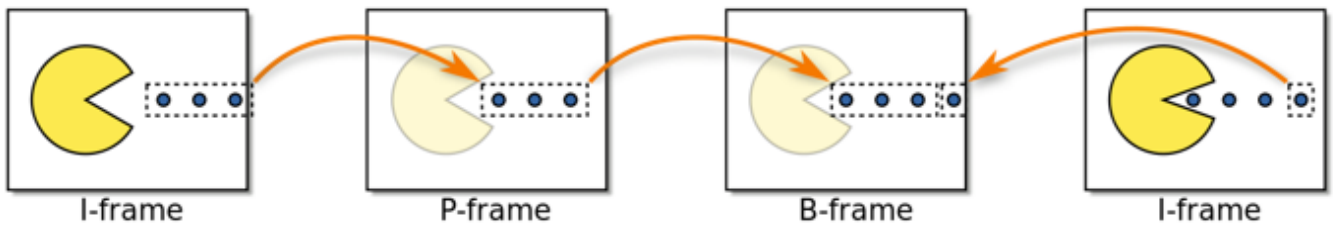
图片 3 评测方法

其中橙色的三角形是官方提供的基准时间和误差，只有优于基准方法的成绩才被视为有效成绩，而其他成绩（黑色三角）则被视为无效成绩。时间和误差会根据基准成绩归一化到 0-1 之间。在有效成绩中，会找出最小误差和最短时间的两个成绩（绿色三角形和红色三角形），然后最小误差和最短时间会组成一个参考点（蓝色圆圈）。最终所有的有效成绩都会和参考点计算距离，距离最短的方法视为优胜。从评测方法分析，时间和精度都是很重要的因素。而时间和精度往往是矛盾的，所以必须进行一定的取舍。

视频解码

因为时间是一个很重要的因素，而视频解码又是一个很费时间的过程，所以如何设计解码模块是本次竞赛中的一个关键。我们采用了多线程软解提取关键帧的方法。

主流的视频编码方式中，每个视频主要包含三种图片帧，分别叫做：Intra-coded frame（I 帧），Predictive frame（P 帧）和 Bi-Predictive frame（B 帧）。其中 I 帧是一张完整的图片。P 帧记录了与之前的帧的差别，所以在解码 P 帧时必须要参考之前的图片帧。而 B 帧不仅需要参考之前的图片帧，还需要参考之后的图片帧才能完整解码。图片 4 阐明了这三个概念 [2]。



图片 4 I 帧，P 帧与 B 帧

显而易见，P 帧和 B 帧的解码是相对较慢的，而直接解码 I 帧则可以获得更快的速度。同时，由于我们需要解码不止一帧，所以我们采用了多线程的方式，每一个线程负责解码一个关键帧。整个解码过程使用 FFmpeg 实现。

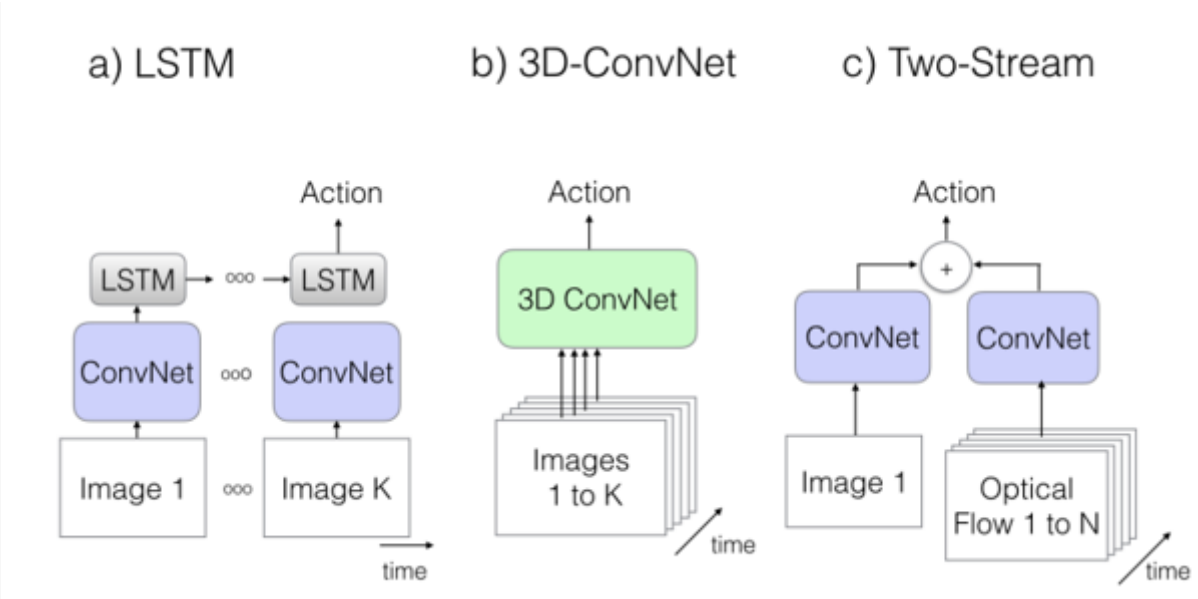
模型设计

解决了解码问题后，接下来的问题在于如何用所得的多帧来进行分类。

- **主流方法**

目前主流的视频分类的方法有三大类：基于 LSTM 的方法，基于 3D 卷积的方法和基于双流的方法。图片 5 展示了这三种框架的大体结构 [3]。

- 基于 LSTM 的方法将视频的每一帧用卷积网络提取出每一帧的特征，然后将每一个特征作为一个时间点，依次输入到 LSTM 中。由于 LSTM 并不限制序列的长度，所以这种方法可以处理任意长度的视频。但同时，因为 LSTM 本身有梯度消失和爆炸的问题，往往难以训练出令人满意的效果。而且，由于 LSTM 需要一帧一帧得进行输入，所以速度也比不上其他的方法。
- 基于 3D 卷积的方法将原始的 2D 卷积核扩展到 3D。类似于 2D 卷积在空间维度的作用方式，它可以在时间维度自底向上地提取特征。基于 3D 卷积的方法往往能得到不错的分类精度。但是，由于卷积核由 2D 扩展到了 3D，其参数量也成倍得增加了，所以网络的速度也会相应下降。
- 基于双流网络的方法会将网络分成两支。其中一支使用 2D 卷积网络来对稀疏采样的图片帧进行分类，另一支会提取采样点周围帧的光流场信息，然后使用一个光流网络来对其进行分类。两支网络的结果会进行融合从而得到最终的类标。基于双流的方法可以很好地利用已有的 2D 卷积网络来进行预训练，同时光流又可以建模运动信息，所以精度往往也很高。但是由于光流的提取过程很慢，所以整体上制约了这一方法的速度。

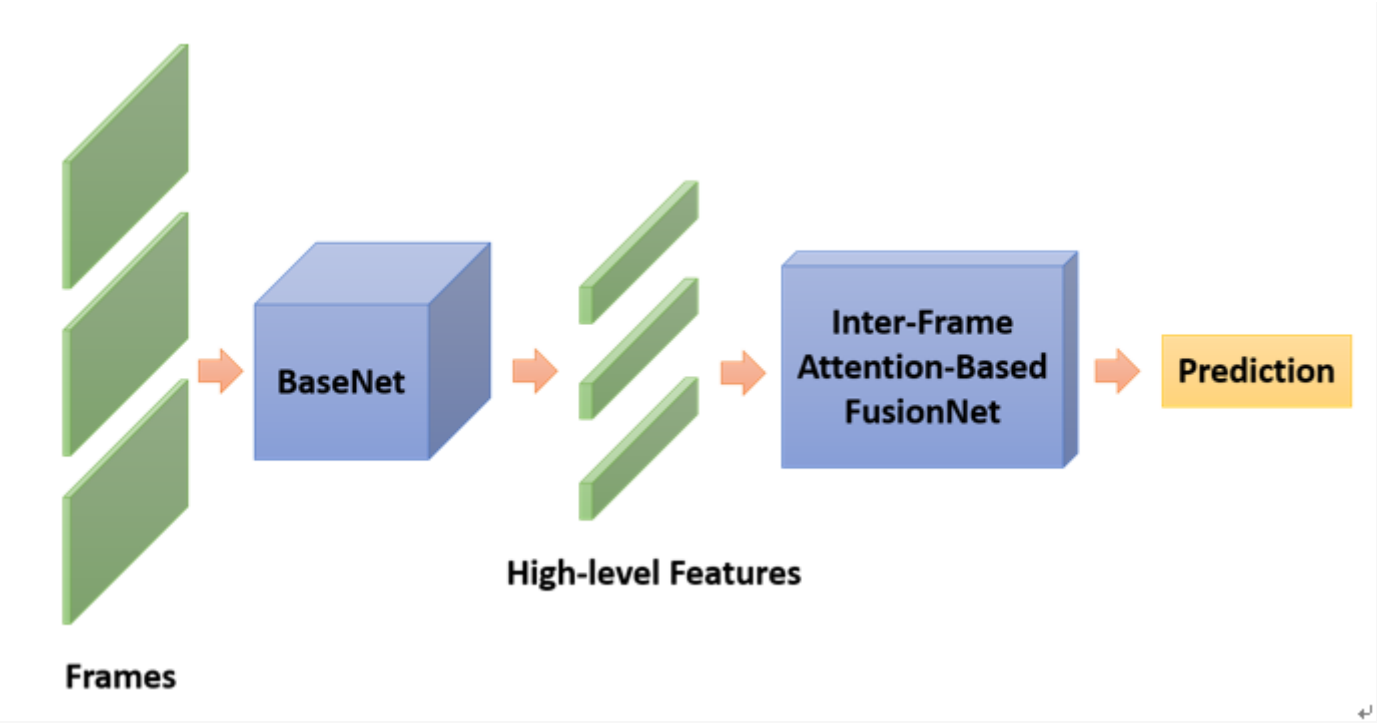


图片 5 主流的视频分类的方法

综上所述，主流的方法都不太适用于短视频实时分类的任务，所以我们特别设计了一个适用于短视频实时分类的框架。

- **我们的方法**

图片 4 展示了我们的解决方案的整体框架：给定一个视频，我们首先会从中稀疏采样固定数量的图片帧，然后将这些帧组成一个 batch，送入到一个 BaseNet 中。这个 BaseNet 是在已有的 2D 卷积网络基础上优化改进得到的，具有较强的特征提取能力。BaseNet 输出的高层的特征往往具有很强的语义信息，但是却没有时间上的融合。所以我们特别设计了一个基于帧间注意力机制的融合模型，将 BaseNet 提取的不同帧的特征作为一个输入送入融合模型中，最终由融合模型得到预测的结果。由于融合模型比较小，推理速度很快，而且参数量较少，也比较容易训练。整个模型在 mxnet 上进行构建和训练。基于这样的设计，我们的模型可以得到很快的推理速度，同时又不会损失太多精度。



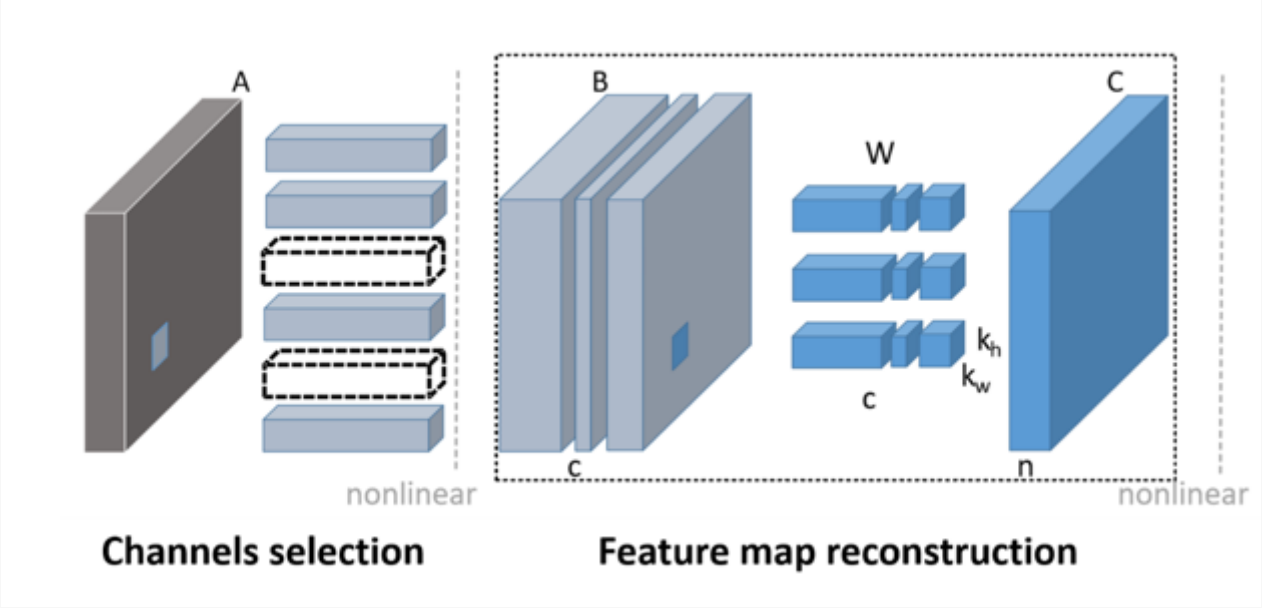
图片 6 整体框架

模型压缩

当有了训练好的模型后，为了进一步提高速度，模型压缩是必不可少的。因为计算平台是 GPU，所以我们使用了两种比较适用于 GPU 的方法：剪枝和量化。

- **模型剪枝**

由于需要在 GPU 上运算，这里我们主要考虑在通道维度的剪枝。假设卷积的参数是具有稀疏性的，我们剪掉其中一些不重要的参数，网络仍然可以达到之前的精度。



图片 7 剪枝

剪枝过程分为两步：首先，我们会基于 LASSO 回归来找到每一层中最具代表性的通道，然后将没用的通道去掉，再使用平方差损失微调剪枝后的网络来最小化重构误差。这样的操作会对每一层分别进行，经过几轮迭代后便可以达到不错的压缩效果，同时还可以保证精度不会损失太多。

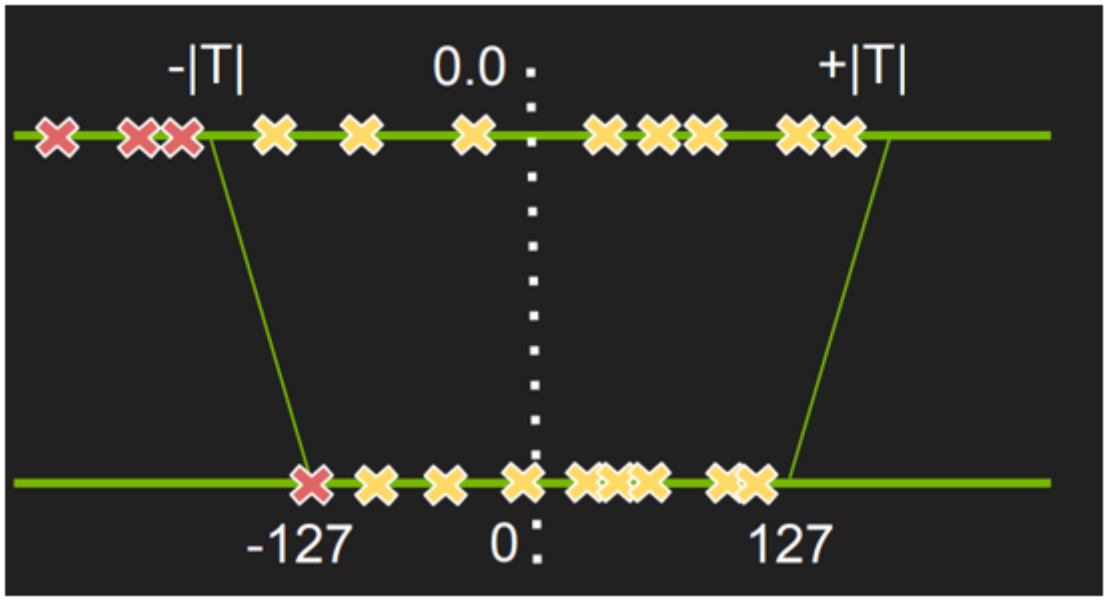
- **模型量化**

由于比赛提供的 GPU 是支持 int8 计算的，所以我们考虑将原来的基于 float32 数据类型训练的模型转换为 int8 的数据形式进行推断，也就是量化操作。这里我们采用的比较简单的线性量化，也是 TensorRt 中使用的方法 [4]。

$$\text{Tensor Values} = \text{FP32 scale factor} * \text{int8 array}$$

图片 8 线性量化

假设每个张量的数据符合均匀分布，那么其中的每一个元素就可以表示为一个 int8 数和一个 float32 的比例因子相乘的结果。比例因子是对于整个数组共享的。这样在张量间进行相乘运算时就可以先进行 int8 的计算，最后再统一乘上比例因子，从而加快运算。那么接下来的问题在于如何确定比例因子，比例因子的作用是将原始张量的数值范围映射到-127 到 127（int8 的数值范围）。由于大多数情况数据并不是完全的均匀分布，所以直接映射会造成精度损失。



图片 9 基于阈值的线性映射

为了解决这个问题，TensorRt 中会对每一层的数据分布进行统计，然后根据得到的分布确定一个阈值（如图片 9）。在映射的过程中，阈值之外的数会被统一映射到-127 和 127 之 间，阈值之内的数据会假设为一个均匀分布然后进行映射。这样就可以保证在加快速度的同时也不至于有较大的精度损失。

总结

我们的解决方案可以归纳为三个部分：视频解码部分，我们采用了多线程提取 I 帧的方式。模型设计部分，我们采用了稀疏采样与帧间注意力融合的方法。模型压缩部分，我们采用了通道剪枝和量化的方法。最终我们的解决方案在测试集上的速度为平均每个视频 58.9ms，精度为 87.9%。

参考文献

[1] 「AI Challenge | Introduction.」[Online]. Available: https://challenge.ai.meitu.com/mtsvrc2018/introduction.html. [Accessed: 21-Nov-2018].

[2] 「視訊壓縮圖像類型,」维基百科，自由的百科全书. 08-Jul-2018.

[3] J. Carreira and A. Zisserman,「Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,」in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[4] S. Migacz,「8-bit Inference with TensorRT.」[Online]. Available: http://on-demand.gputechconf.com/gtc/2017/presentation/s7310-8-bit-inference-with-tensorrt.pdf