《Flocking-based Live Streaming of 360-degree Video》论文学习汇报

CUC - 19 - 数媒技 - 杨雪婷

2021.01.28

汇报内容介绍

- 1. 问题
- 2. Fov预测
- 3. 用户群体划分
- 4. 缓存策略
- 5. 实验及指标

Part One: 问题

现有研究的不足:

1、Fov预测的不足:

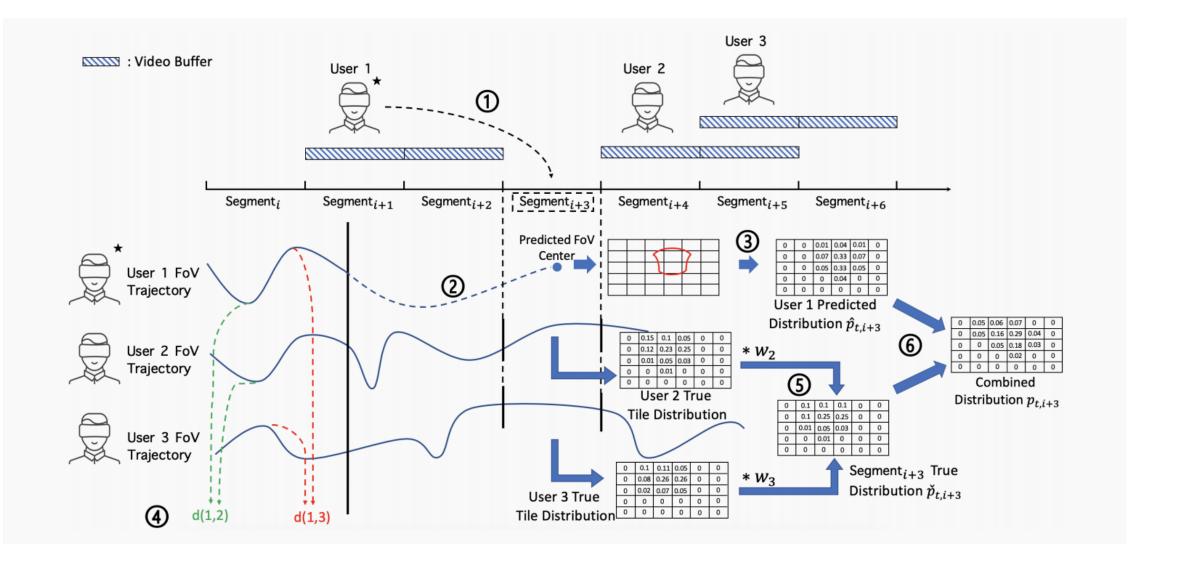
- (1) 主要是基于用户自身历史数据的预测
- (2) 协同预测只有针对VoD的研究
- (3) 假设已有大量的历史用户群体

2、 缓存的不足:

(1) 大量的请求和冗余的流量

等....

Part Two: Fov 预测



基于自身的数据预测:

3中: Kalman Filter(卡尔曼滤波器)过滤噪声后,带入Fov预测方法(截断线性FoV预测)

0	0	0.01	0.04	0.01	0
0	0	0.07	0.33	0.07	0
0	0	0.05	0.33	0.05	0
0	0	0	0.04	0	0
0	0	0	0	0	0

User 1 Predicted Distribution $\hat{p}_{t,i+3}$

得到中心C后,我们利用 ERP投影,计算出每块贴图中有多少在预测Fov中(百分比)。

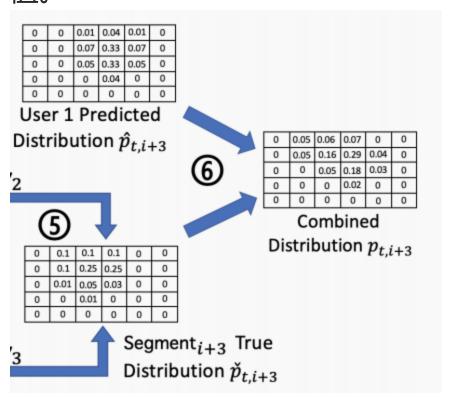
基于真实的数据预测:

- 平均大圆球距离计算d(1,2),d(1,3)
- 计算**权重W**: (这里的γ和Q是离线得到的最优值,分别为3和0.5)

$$w_j = \mathcal{H}(d(j,t)) = \frac{e^{-\gamma(d(j,t)-\phi)}}{1 + e^{-\gamma(d(j,t)-\phi)}},$$

将两矩阵叠加:

最后我们将用户自身预测的结果(^)和通过历史数据预测的结合起来成为最终的Fov预测值。



这里的两个矩阵叠加, 也分别有相应的权重:

$$\tilde{p}_{t,i+3} = \alpha \hat{p}_{t,i+3} + (1-\alpha) \check{p}_{t,i+3}.$$

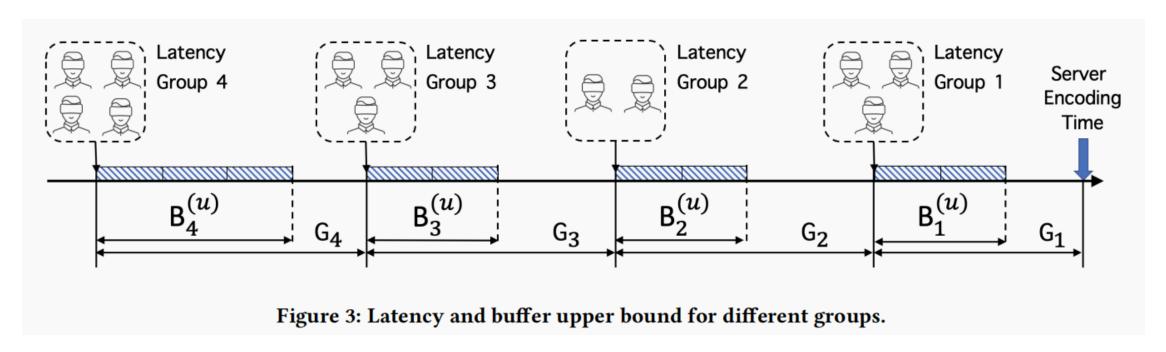
其中的 α 是以下的算式得出,我们可以发现当在此用户之前如果没有其他用户观看带预测视频的时候, $\alpha = 1$,也不会影响整个算法的运行。

$$\alpha = \frac{1}{1 + \sum_{j \in S_t} w_j}$$

以上应该是在服务器中下载信息后,客户端计算。

Part Three: 用户群体区域划分

这部分我们需要确定让flock中的哪些用户在flock的较前面, (他们的概率矩阵不那么准确, 因为在他们之前的数据非常少), 哪些用户在flock的较后面。



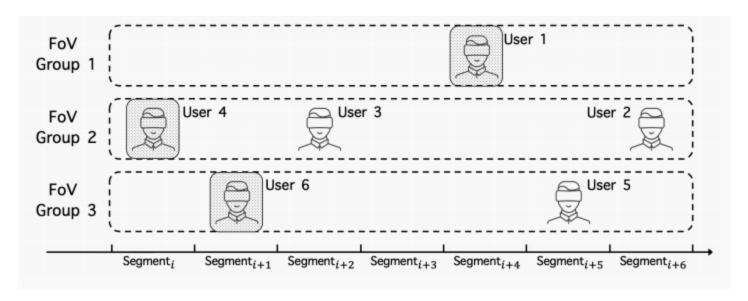
在实验中:

计算用户带宽的相对标准偏差(RSD)。通过将其与预定义的RSD阈值进行比较,可以对用户的网络状况进行分类,从而将用户分配到相应的延迟组

Part Four: 缓存策略

Latency-FoV based Caching Strategy (基于延迟Fov的缓存策略)

基于延迟fov的缓存策略。根据**用户过去的视场轨迹相似度**,将用户分为不同的视场组。 (这个是动态分组的,每5s重新计算一次)



标记:将标记具有**较长的延迟**和共享相似观看行为的**邻居较少**的用户:(用户4和6)被标记,组中只有一个用户(用户1)也被标记。

- 标记用户的请求被忽略
- 其他tile依据LRU caching rule请求

Transcoding at edge server

另一种方法(设想):假设有充足的缓存存储和带宽。

- 任何一段视频准备好,就以最高质量缓存下来。 (U-transcoding)
- 任何一段视频**被请求**,就以最高质量缓存下来。 (E-transcoding) 边缘服务器可以操作**从高编码率到请求的编码率**的**实时转码**。这样可以节省在核心网络上传输的流量。

Part Five: 评价指标

Fov预测评价 -- KL散度

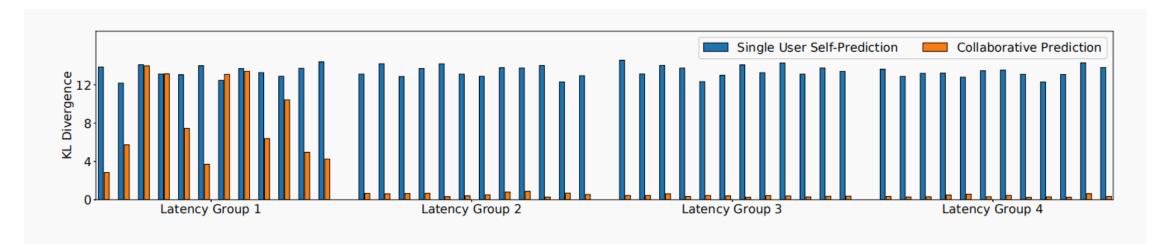
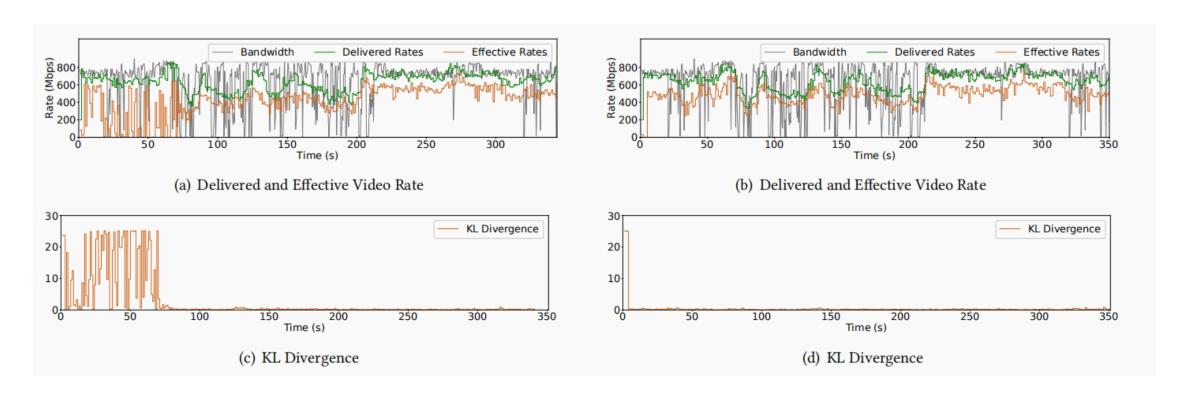


Figure 6: KL Divergence between predicted tile attention distribution and ground truth distribution of all users in different latency groups. Latencies and buffer upper bounds of group 1 to 4 are {3s, 2s}, {8s, 3s}, {13s, 3s} and {19s, 3s}, respectively.

我们根据用户的网络环境分成几段,第一梯队的属于in the front of flock因此含有的协作预测信息太少,所以橙色条没有后面明显。

但是可以发现,加上了协同预测后的所有KL散度都减小甚多,说明协同预测非常有效。

智能分配群区域 -- 有效视频质量 / Freeze



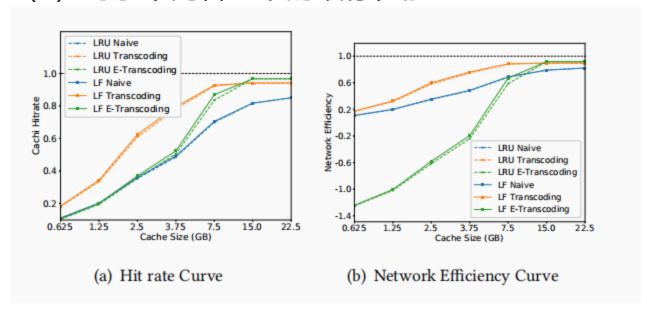
灰色线是有效带宽, (a),(b)图中的绿色线(交付视频比特率)大致相同, 但是橙色线(有效视频比特率)明显(b)比(a)强。

因为有效视频和对Fov的预测有关,因此右面的计算后的智能分配会使得他有更多的协作用户数据,因此(d)比(c)的KL散度更准确些,于是Eeffective Rates 也就更高。

缓存性能

- (1) 对于无限制的缓存容量 -- 编码的影响
 - cache hit rate
 - network efficiency
 - transcoding ratio
 - the storage requirement

(2) 对于不同容量下的缓存性能



当容量达到15G左右时基本到达峰值,可以覆盖48s的内容。

Network Efficiency: A-B/B

A: 用户请求的

B: 从原始服务器传送到缓存的总流量

实验

- 9视频48用户的轨迹
- WiGig模拟网络状况
- HMM合成带宽轨迹 (page8)
- 码率选择 (?
- 预测带宽: 计算过去10个视频段下载带宽的谐波均值作为下一个视频段的预测带宽。