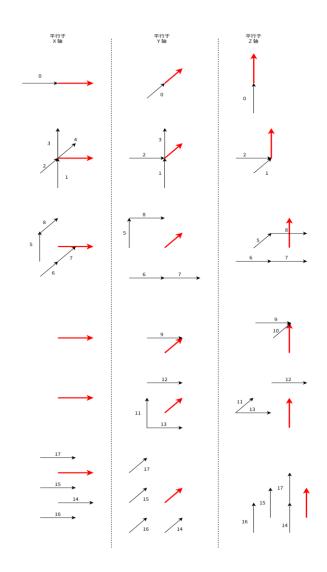
测 18 条边熵



数据统计

未分方向测试标识信息熵值

		18 条边中某条边未被占 据	18 条边中某条边被占据		
当前待编码 边未被占据	0	P00	P10		

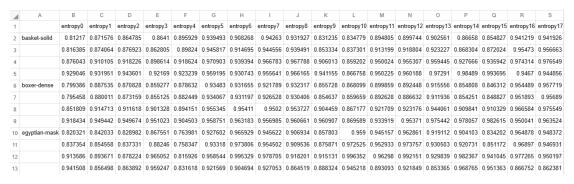
当前待编码 边被占据	1	P01	P11
	entropy	E1	E2

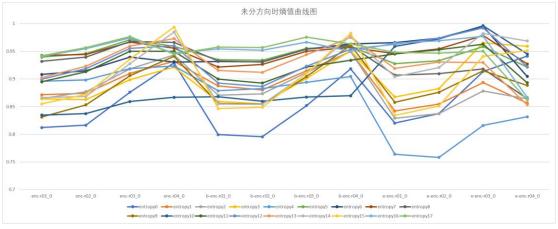
按公式:

$$E = \sum rate * \sum P \log_2 \frac{1}{P}$$

其中rate表示 18条边中某条边被占据和未被占据的概率。

每个序列每个码率点求得一个平均值

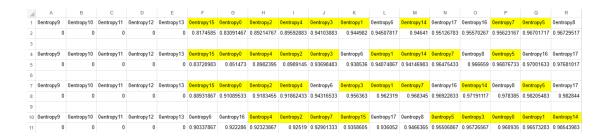




分方向测试标识信息熵值

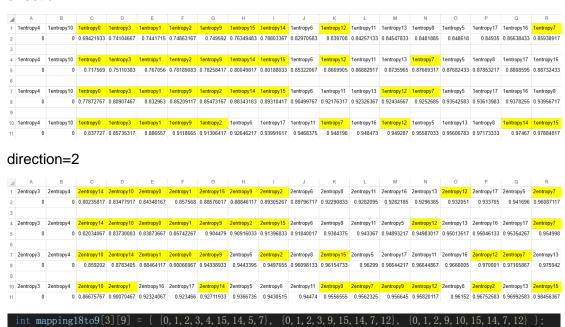
- 标黄表示现有使用的9条边
- 每一行按<mark>熵增顺序</mark>排列

direction=0



direction=1

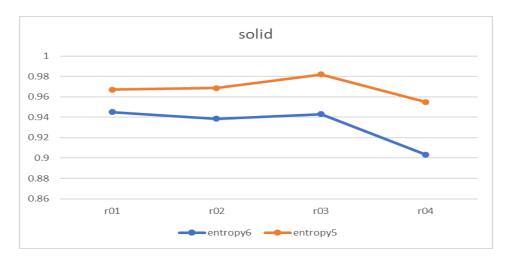
适的:



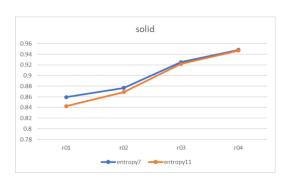
• 基本上是使用熵值较小的边,但最后使用的不相邻的垂直与平行边熵值不是最合

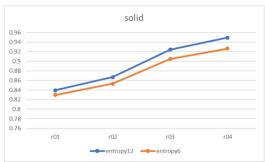
direction	不合适的边	合适的边	
0	5	6	
1	7、12	11、6	
2	7、12	6、11	

- 依据上述数据,我打算修改各个方向使用的边,用熵值较小的替换熵值较大的边, 以寻求性能增益。
- 1. direction=0 时——用 entropy6 替换 entropy5

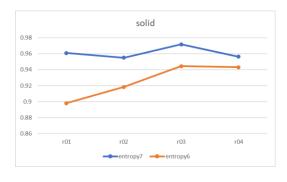


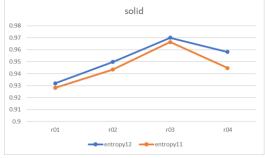
2. direction=1 时——用 entropy11 替换 entropy7、用 entropy6 替换 entropy12





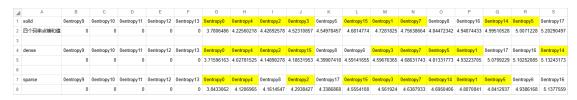
3. direction=2 时——用 entropy6 替换 entropy7、用 entropy11 替换 entropy12





分方向测试位置信息熵值

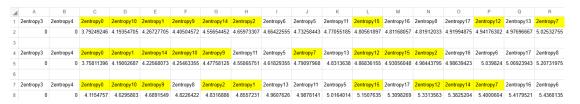
- 与优化标识信息所使用的边流程相同,对位置信息进行熵值计算与分析
- 1. direction=0



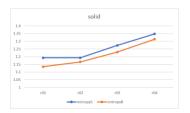
2 direction=1

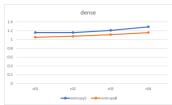


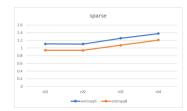
direction=2



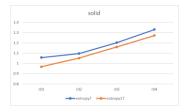
- 分析数据得到以下待修改边与修改边的熵值对比折线图:
- 1. direction=0——用 entropy8 替换 entropy5

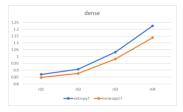


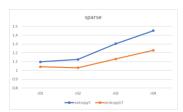




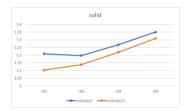
2. direction=1——用 entropy17 替换 entropy7

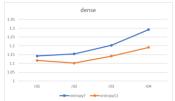


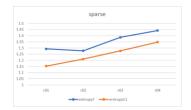




3. direction=2——用 entropy11 替换 entropy7





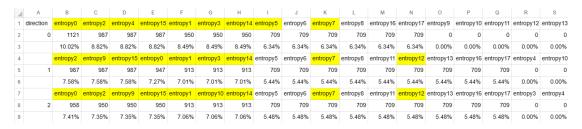


不同方向上 18 条边分别存在数量在总存在边数中的占比

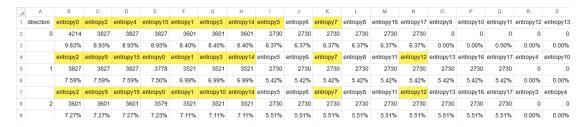
为了考虑各个边在进行上下文构建时的所占权重, 我对 18 条边的存在数量占比进行了统计, 替换前熵值较大的边与替换后熵值较小的边, 两者在 18 条边中的存在数量占比是一致的, 因此不会因为占比问题影响熵编码效果。

solid 类型 basketballplayer 序列下测试:

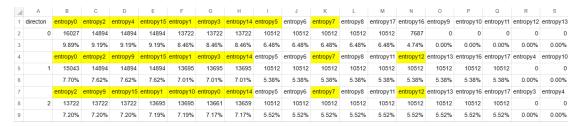
R01 码率点



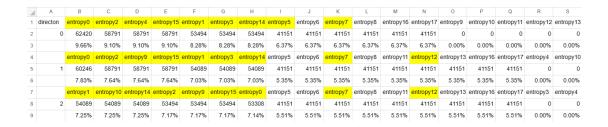
R02 码率点



R03 码率点



R04 码率点



性能测试

修改方案

标识信息部分: 位置信息部分:

direction=0——5->6 5->8

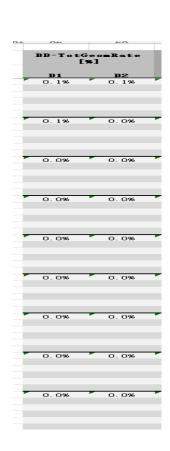
direction=1——7->6 7->17

direction=2——7->6 7->11

测试结果

solid 类型:

Class	Sequence
solid	basketball_player_vox11_00000200
	basketball player vox11 00000200 dancer vox11 00000001
	dancer_vox11_00000001 dancer_vox11_00000001
	dancer_vox11_00000001
	dancer_vox11_00000001
	dancer vox11 00000001
	dancer vox11 00000001
	facade_00064_vox11
	facade_00064_vox11
	facade_00064_vox11
	facade_00064_vox11 facade_00064_vox11
	facade 00064 vox11 longdress_vox10_1300
	longdress_vox10_1300
	longdress_vox10_1300
	longdress_vox10_1300
	longdress_vox10_1300
	longdress vox10 1300
	loot_vox10_1200
	loot_vox10_1200
	loot_vox10_1200 loot_vox10_1200
	loot_vox10_1200
	loot vox10 1200
	queen_0200
	queen 0200
	redandblack_vox10_1550 redandblack_vox10_1550
	redandblack_vox10_1550
	redandblack_vox10_1550
	redandblack_vox10_1550
	redandblack vox10 1550
	soldier_vox10_0690
	soldier vox10 0690 thaidancer viewdep vox12
	thaidancer_viewdep_vox12 thaidancer_viewdep_vox12
	thaidancer_viewdep_vox12 thaidancer_viewdep_vox12
	thaidancer_viewdep_vox12
	thaidancer_viewdep_vox12
	thaidancer viewden vox12



		lossy geometry, lossy attributes [all intra]					
C2_ai		End-to-End BD-AttrRate [%]			Geom. BD-To	Geom. BD-TotGeomRate [%]	
	Luna	Chroma Ch	Chroma Cr	Reflectance	D1	D2	
Solid average	0.0%	0.0%	0.0%	\$2.0W	0.0%	0.0%	

问题:为什么将使用的边换成熵值较小的边以后,熵编码效率没有得到提高?