Slice buffer design

Table of Contents

1. Overview for new design 2

1.1 Origin design 2

1.1.1. single thread 2

1.1.2. Multi thread 3

1.2 Current design in review 4

# 1. Overview for new design

## 1.1 Origin design

SSlice\* pSliceInLayer // slice buffer for all slices in layer

### 1.1.1. single thread

pSliceInLayer

…

…

…

Slc3

Slc2

Slc1

Slc0

**reallocate** when current slice index larger than max slice num

…

pSliceInLayer

Slc2

Slc1

Slc0

Slc0

Slc1

Slc2

…

…

…

…

### 1.1.2. Multi thread

pSliceInLayer

Slc0

Slc1

Slc2

Slc3

…

…

Thrd0

Thrd1

**reallocate** when current slice index larger than max slice num ,

--step 0: thread[0] detect that current slice index larger than max slice num;

--step 1: thread[0] need to wait thread[1] completed current slice encoding task

-- step 2: thread[1] stop slice encoding and thread[0] reallocate slice buffer,

--step 3: thread[0]/thread[1] start to encode new slice

pSliceInLayer

Slc0

Slc1

Slc2

...

**Step 0/1**

Thrd1

Thrd0

Thrd0

pSliceInLayer

Slc0

Slc1

Slc2

…

…

**Step 2**

Slc0

Slc1

Slc2

...

…

…

…

**Step 3**

Thrd2

Thrd0

## 

## 1.2 Current design in review

SSlice\* pSliceInLayer; // the same with origin design, single thread only

// will not allocate when multi thread on

SSlice\*\* ppSliceInLayer; // point to actual slice buffer

//based on thread mode

SSlice\* pSliceInThread[MaxThreadNum]; // multi Thread only

// will not allocate for single thread

for reallocate, each thread will do it independently, and will update ppSliceInLayer when all slices in layer all encoded.

Slc0

Slc1

Slc2

Slc3

Slc4

…

…

pSliceInLayer

Multi thread

Single thread

pSliceInThread[1]

…

…

Slc5

Slc3

Slc1

…

…

Slc0

Slc2

Slc4

pSliceInThread[0]

ppSliceInLayer

…

…

Slc4

Slc3

Slc2

Slc1

Slc0