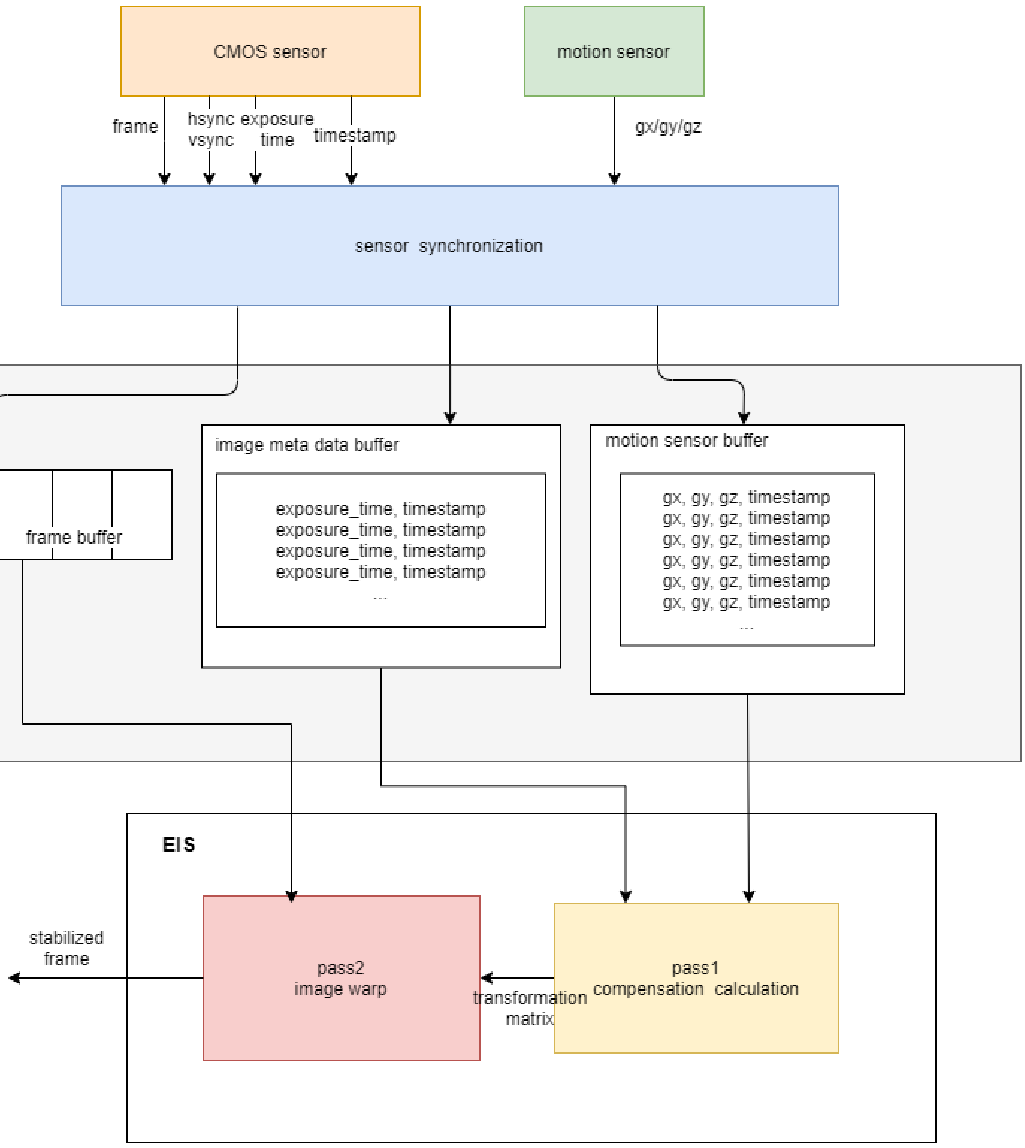
Development Guide of an EIS System

An Electronic Image Stabilization utilizes electronic motion sensor, normally gyroscope, to estimate device motion and compensate jitter to stabilize a video.

Main components of an EIS system includes:

- Sensors
  - CMOS sensor
  - motion sensor
- Sensor synchronization
- Queue handler
- Image/Video Stabilization algorithm



The Requirements of an EIS system

To develop an EIS system, we would handle matters about sensor calibration and synchronization, since the EIS applies on a real-time system, a simulator is very important which can simulate a real-time system on device to mimic the EIS function which run on a device. I write down some requirements and works which have to do in developing an EIS system according to my previous experience.

Sensor Calibration

For most MEMs Gyroscope sensors drift with bias, which accumulate errors by time. For those application of video stabilization, which apply low-pass filter to remove unwanted jitter from video, some weired drift would occur with uncalibrated gyro sensor. The gyro sensor would calibrated both static and temperature bias.

Here are some technical articles about gyro bias calibration:

[Gyro Mechanical Performance: The Most Important Parameter]

[A SIMPLE CALIBRATION FOR MEMS GYROSCOPES]

Video Meta Data

For a perfect EIS system which stabilized not only inter-frame motion (the motion between frame and frame) but also intra-frame motion(the motion during a frame). So that we require the time duration of time of adjacent frames generalized, and the the each time of the line which generalized during a frame to compensate rolling shutter effect.

The required video meta data are:

- Focal length
- Exposure time
- CMOS blank time
- Vsync duration (frame rate)
- Hsync duration

Focal length calculation:

The focal length *in pixel* is for a camera intrinsic matrix,

$$k = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- focal_length_x_in_pixel = (focal_mm / sensor_width_mm) * full_frame_width_in_pixel
- focal_length_y_in_pixel = (focal_mm / sensor_height_mm) * full_frame_height_in_pixel

Calculation of duration between frame-and-frame:

```
center_line_no = blanking + int(param.effective_height / 2)
prev_frame_center_ts = vsync_timestamp - (
    param.frame_height_in_pixel - center_line_no) * hsync_duration -
prev_frame_exposure_time / 2
this_frame_center_ts = vsync_timestamp + vsync_duration - (
    param.frame_height_in_pixel - center_line_no) * hsync_duration -
this_frame_exposure_time / 2
```

CMOS blank time and Exposure time

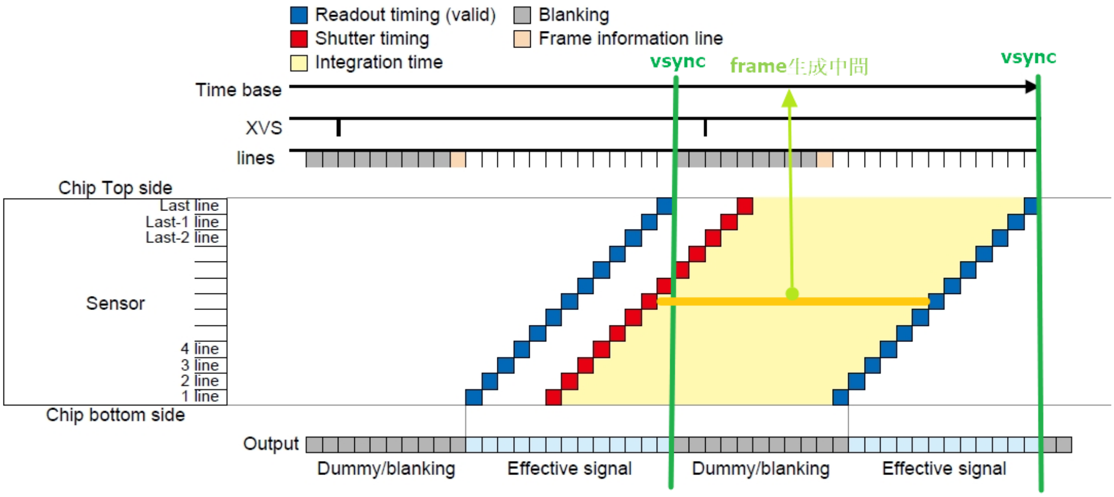


Image Drawing of Shutter Operation

Blank time measurement

The number of blank line could be found in CMOS sensor specification, or the blank line duration could be measured by oscilloscope.

Sensor Synchronization

An EIS system stabilizes video frames by eliminate jitter which calculated from gyroscope. There are two kind of sensors, CMOS and motion sensor, in an EIS system. The sensor synchronization is extremely strict for rolling shutter correction. Normally, there is no clock system in most motion sensor, however, for a CMOS sensor with Hsync and Vsync signal can be utilized to synchronized the time with gyroscope.

There are some kind of sensor logs carefully designed for synchronization time with CMOS sensor:

gyro log:

The order from left to right of each column:

1. gyro sample timestamp (us)
2. current vsync timestamp (us)
3. gyro x-axis (degree/sec)
4. gyro y-axis (degree/sec)
5. gyro z-axis (degree/sec)
6. is this sample vsync signal or not
7. hsync count number

imu.log

```
000045273853 000045244435 -0.610352 +0.549316 -0.122070 00000 03143
000045275854 000045244435 +0.305176 +0.244141 +0.427246 00000 03356
000045279852 000045277731 -0.671387 +0.122070 +0.427246 00001 00235
000045281853 000045277731 +0.549316 +0.793457 +0.305176 00000 00448
000045283854 000045277731 -0.732422 +0.000000 +0.000000 00000 00661
000045285852 000045277731 +0.671387 -0.366211 +0.915527 00000 00874
000045287852 000045277731 -0.671387 -0.366211 -0.244141 00000 01087
000045289852 000045277731 +0.244141 +0.915527 +0.610352 00000 01299
000045291854 000045277731 -0.122070 +0.793457 +0.061035 00000 01512
000045293852 000045277731 -0.488281 -0.061035 +0.183105 00000 01725
000045295852 000045277731 +0.610352 -0.915527 +0.305176 00000 01938
000045297852 000045277731 -0.915527 -0.366211 +0.061035 00000 02150
000045299854 000045277731 +0.793457 +0.854492 +0.854492 00000 02363
000045301852 000045277731 -0.854492 +0.000000 -0.427246 00000 02576
000045303852 000045277731 +0.122070 -0.732422 +0.549316 00000 02789
000045305852 000045277731 -0.061035 +0.000000 +0.183105 00000 03001
000045307854 000045277731 -0.244141 +0.915527 +0.000000 00000 03214
000045309854 000045277731 +0.427246 +0.549316 +0.488281 00000 03427
000045311852 000045311033 -0.793457 -0.366211 +0.183105 00001 00095
000045313852 000045311033 +0.610352 -0.122070 +0.732422 00000 00307
000045315853 000045311033 -0.671387 +0.610352 -0.305176 00000 00520
000045317854 000045311033 +0.122070 +0.732422 +0.854492 00000 00733
000045319852 000045311033 -0.244141 -0.122070 +0.000000 00000 00946
000045321852 000045311033 -0.244141 -0.305176 +0.122070 00000 01158
000045323853 000045311033 +0.305176 +0.427246 +0.549316 00000 01371
000045325854 000045311033 -0.732422 +0.061035 +0.244141 00000 01584
000045327852 000045311033 +0.793457 -0.610352 +0.366211 00000 01797
000045329852 000045311033 -0.610352 -0.427246 -0.061035 00000 02010
000045331853 000045311033 +0.366211 +1.342773 +0.854492 00000 02222
000045333854 000045311033 -0.122070 +0.732422 -0.305176 00000 02435
000045335852 000045311033 -0.244141 -0.854492 +0.244141 00000 02648
000045337852 000045311033 +0.183105 -0.549316 +0.549316 00000 02861
000045339852 000045311033 -0.549316 +0.549316 +0.122070 00000 03073
000045341854 000045311033 +0.732422 +0.976562 +0.610352 00000 03286
```

CMOS log:

The order from left to right of each column:

1. vsync timestamp (ns)
2. exposure time (ns)

cs.log

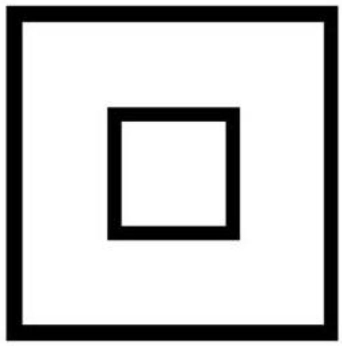
```
045479437598 000003873802
045512759598 000003873802
045546085598 000003873802
045579422598 000003873802
045612743598 000003873802
045646050598 000003873802
045712701598 000003873802
045779347598 000003873802
045812670598 000003873802
045879320598 000003873802
046279188598 000003873802
046312517598 000003873802
046345855598 000003873802
046379157598 000003873802
046412484598 000003873802
046445804598 000003873802
046479141598 000003873802
046512446598 000003873802
046545779598 000003873802
046579099598 000003873802
046612451598 000003873802
046645741598 000003873802
046679065598 000003873802
046712398598 000003873802
```

Synchronization Validation

The synchronization validation is to verify the time alignment between the video frames and motion sensor. Test video and gyro sensor data are record in specific motion pattern and frame pattern to easily estimate the motion in a frame.

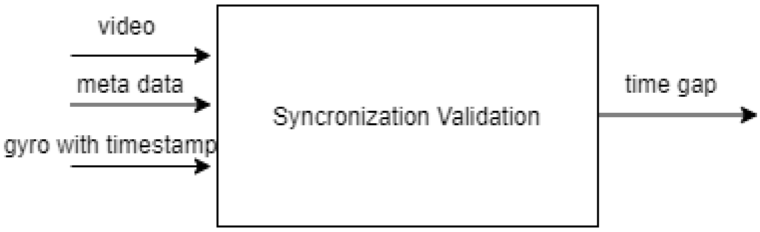
The motion pattern can be record by vibrator to generate some period sine wive motion which make pixels of frames moving in this kind period motion.

Some sort of image pattern are suitable for aligning the time between motion sensor and frame, like square or circle, a square pattern is much more easy to locate four corners and calculate the deformation from image.



The input of the synchronization validation function are video with meta data and gyro data with timestamp. The output of the synchronization is time gap between video and gyro sensor.

If the time gap is fixed from every test set, we can say that the time system of CMOS and gyro sensor are well alignment.



Sensor Queue Handler

There are three queue buffer in an EIS system:

- video meta data buffer

  store Vsync timestamp and exposure time
- motion sensor buffer

  store gyro data with timestamp
- frame buffer

  store video frames

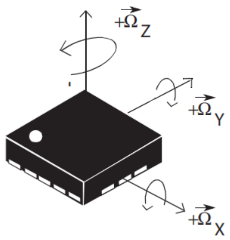
The CMOS sensor pushes the video meta data and frame into *video meta data buffer* and *frame buffer*. The gyro sensor pushes the gyro data with timestmap into *motion sensor buffer*. The video stabilization function pops these buffers data to generate an stabilized frame. The size of frame buffer and video meta data buffer are decided by the kernel size of the low-pass filter in the EIS system or by the number of frame of multi-frame denoise. Normally , we buffer the two to three frame time duration gyro data into motion sensor buffer. For example, as a 30 fps frame sample rate, and 500 samples/sec gyro sample rates, the motion sensor buffer with 3 frames duration is:

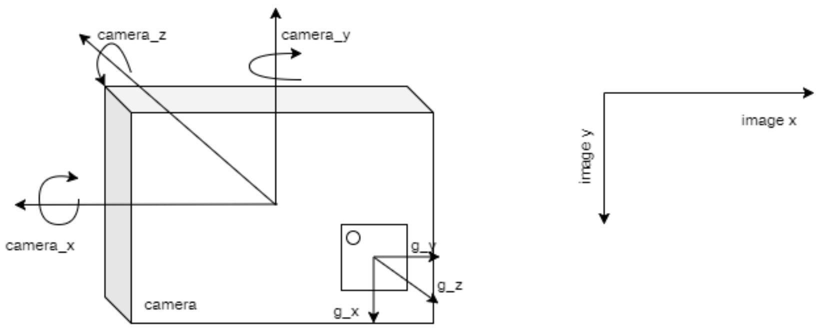
```
500 (samples/sec) /30 fps * 3 (frames buffer) = 50
```

Coordinate System

For a gyroscope coordinate system, there is a mark point on the top, which is used to define the 3-axis coordinate, and angular rate is positive as rotate clockwise, negative vice versa. There would be a coordinate transformation from gyro sensor coordinate system to camera coordinate system.

Here are an example of gyro sensor and camera coordinate system:

<table>
<tr><th>gyro sensor coordinate system</th><th>camera coordinate system</th></tr>
<tr><td>


</td><td>


</td></tr>
</table>

and the coordinate system transformation matrix from gyro to camera system:

$$\begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}$$

You can refer the gyroscope coordinate system from manufacture data sheet:

[L3GD20H MEMS motion sensor: three-axis digital output gyroscope]

, and here is the android sensor coordinate system:

[Android Sensor Coordinate System]

Video Stabilization Function

- EIS pass 1:

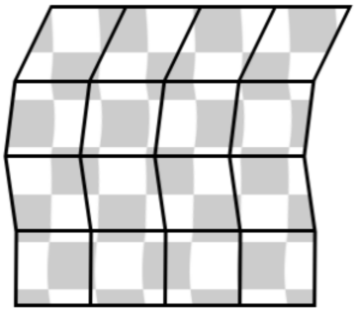
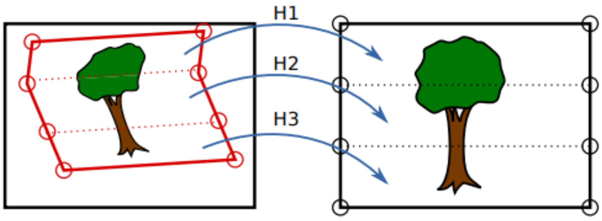
  motion estimation and jitter calculation

- EIS pass 2:

  jitter compensation and image warp

There are normally two main steps in a video stabilization function. They can generally be called EIS pass 1 and pass 2 in EIS. In the first pass, device movement and trajectory are calculated from motion sensor or from frames. And smooth filter or some optimization is apply to the device trajectory, and get the motion vector from the difference between the smooth and real device trajectory. The compensated motion vector or transformation matrix will be the input of second pass.

EIS pass 2 do image warping to compensate the jitter by transformation matrix from EIS pass 1. The number of the transformation matrix used for compensation the jitter is based on how much frequency of the movement of the device. There are two main kind image warp method, grid and line-based:

grid-based image warp	line-based image warp
	

Here are some EIS methods suitable for mobile device application:

[Digital Video Stabilization and Rolling Shutter Correction using Gyroscopes]

[A Non-Linear Filter for Gyroscope-Based Video Stabilization]

Performance Evaluation

The most common method to evaluate EIS performance is to compare stabilized video side-by-side from different devices. However, to have an objective method to evaluate how much level the video be stabilized, like have a stabilized score to compare the before and after stabilization level in the same video.

There are some performance evaluation methods:

- Stabilized score before/after video stabilization

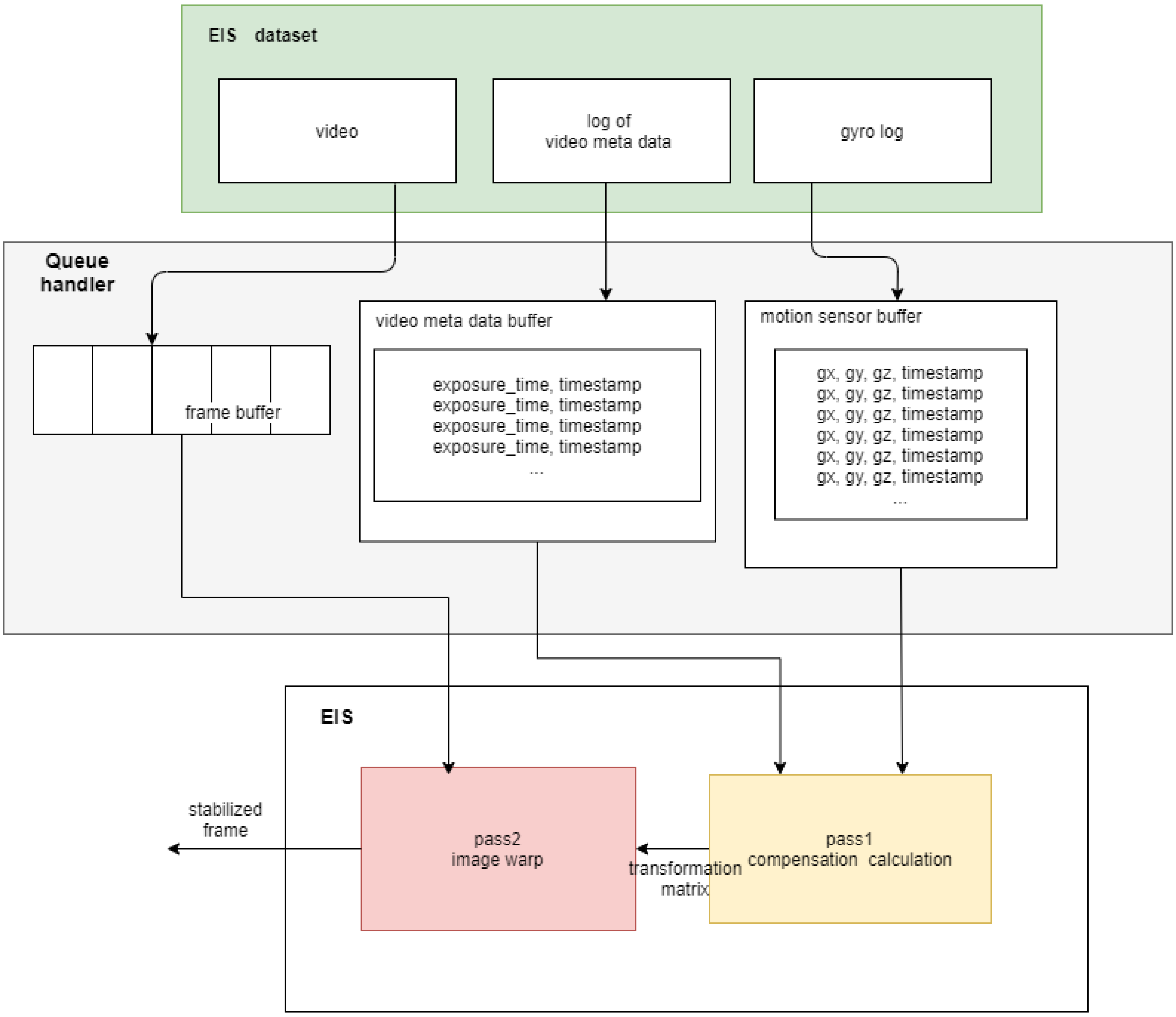
  the score to compare how much stabilized level between origin and stabilized video.
- Stabilization comparison between devices
  - subjective evaluation

    A equipment or a scheme can record two device stabilization video simultaneously. The stabilized videos can be compared side-by-side.
  - objective evaluation
    - DxO
    - stabilized score

      compare stabilized score from different device's videos which record in the same scene(in lab)

EIS System Simulator

EIS algorithm applies in a real time system, so that we would like to have a simulator which can simulate the EIS function in a device. The input of simulator is an EIS dataset which includes, a video, video meta data log, and gyro log. The simulator would output these sensor signal from these logs sequentially by timestamps from these logs, which mimic the CMOS and gyro sensor output the signal samples.



Debug and Logger

The requirements of the logger of EIS system are:

- log with synchronized CMOS and gyro sensor
- as the input of an EIS simulator
- log on-line without drop signal
- used for debug on-line EIS function, which can record both stabilized video and origin video simultaneously. Besides recording stabilized video, we can record preview with origin video.