Emily ChiRockchip Color Optimization Guide ISP2x

ID: RK-SM-YF-603

Release Version:V1.2.1

Release Date: 2021-03-01

Security Level: □Top-Secret □Secret □Internal ■Public

Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian,PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Foreword

Overview

This article aims to describe the debugging of color-related modules, mainly to provide help for engineers who use RkAiq module for image color tuning.

Product Version''

Chip Name	Kernel Version
RV1126/RV1109	Linux 4.19

Target Audience

This document (this guide) is mainly applicable to the following engineers:

ISP debugging engineer

Revision History

Version Number	Author	Revision Date	Revision Description	Corresponding Tool Version
V1.0.0	Wendy Weng Emily Chi	2020-07-30	Initial version	RKISP2.x_Tuner_v0.1.0 and above
V1.1.0	Emily Chi	2020-09-29	Modified to markdown file Add manual white balance/auto white balance parameter description in awb module	Same as above
V1.2.0	Emily Chi	2020-10-19	Fine-tuning the structure Add hdrFrameChoose parameter description to the awb module	Same as above
V1.2.1	Emily Chi	2021-01-04	Corrected many wrong descriptions of awb	Same as above

---

Contents

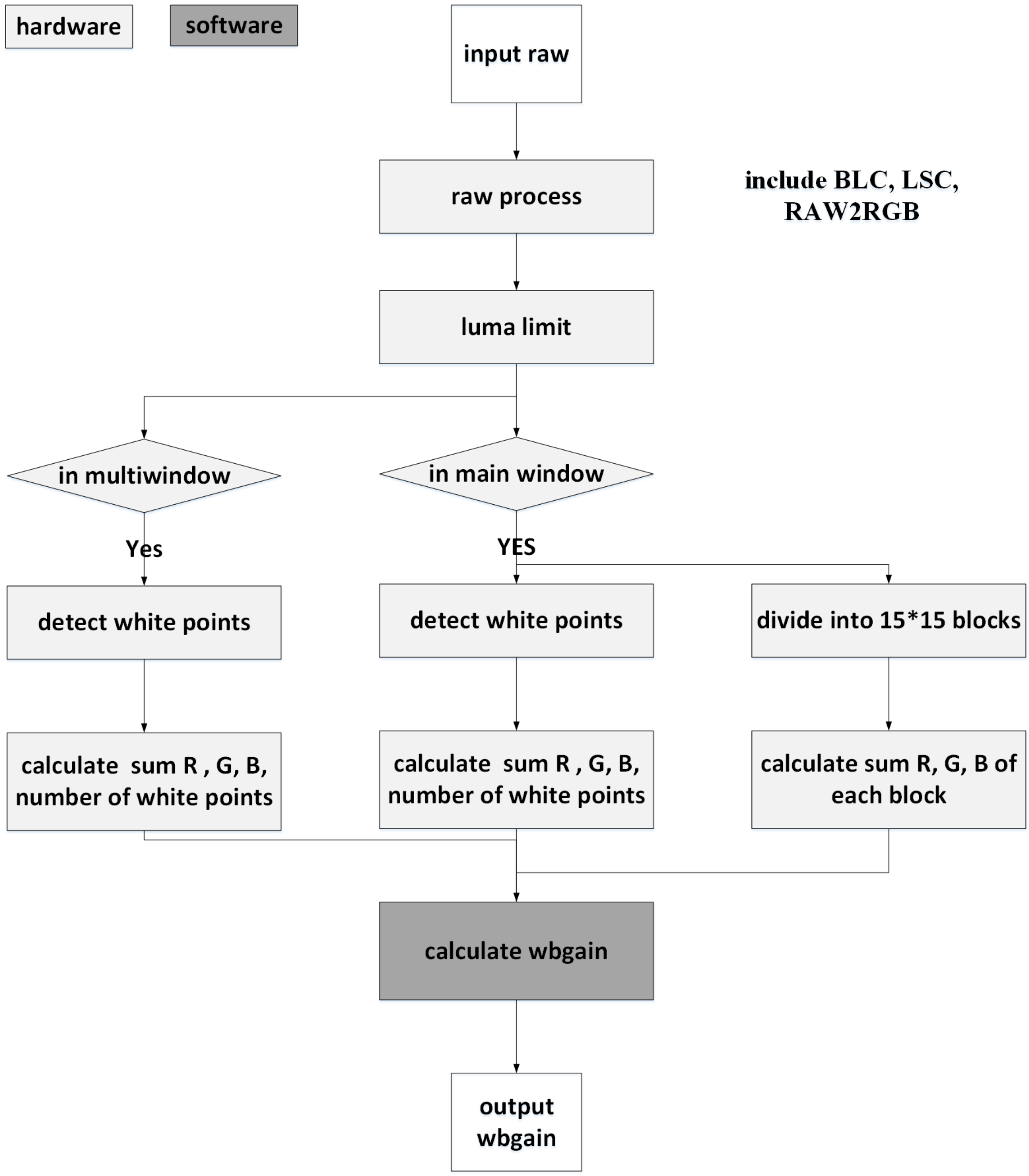
1 Overview

The color adjustment modules related to ISP20 include auto white balance (AWB), color correction (CC), and three dimension look up table (3dlut).

2 AWB

2.1 Function description

The automatic white balance algorithm can automatically calculate the WB gain (white balance gain of the RGB channel) and multiply it with the RGB channel to restore the white affected by the ambient light to pure white, ensuring that the camera will The imaging color is consistent with the true color of the object. When there is a white point in the scene, WB gain is calculated based on the automatically detected white point, and when there is no white point in the scene, the WB gain is obtained by the simple color method. The color adaptation module adjusts the target of white balance correction so that the image after white balance correction is as consistent as possible with the appearance perceived by the human eye. The tone adjustment module adjusts the overall tone according to preferences. The automatic white balance is composed of hardware statistics and software strategies, as shown in the AWB flowchart



AWB process

2.2 Key parameters

Manual white balance/auto white balance/white balance correction enable

Automatic white balance needs to be configured with awbEnable as 1 in xml. The enabling parameters of manual white balance/auto white balance/white balance correction in xml are as follows:

Name	Description
wbBypass	Value 0 or 1 0 means no white balance correction is performed 1 means white balance correction is performed, and the white balance gain used is controlled by awbEnable
awbEnable	Value 0 or 1 0 means using manual white balance gain 1 means using automatic white balance algorithm to calculate white balance
lightSourceForFirstFrame	The value is the light source name. When awbEnable is 1, the white balance gain of the light source is used to turn on the first few frames of the camera. When wbEnable is 0, the white balance gain of the light source is used for all frames. , That is, manual white balance gain
standardGainValue	Value range 0-8 is the white balance gain corresponding to a light source

hdrFrameChoose

Name	Description
mode	Value 0 or 1 0 fixed mode 1 automatic mode, which automatically selects which frame is used for white balance statistics.
frameChoose	Valid when mode is 0; Under two-frame hdr: the value is 0 or 1; 0 Choose short frame for white balance statistics; 1 Choose long frame for white Balance statistics; Under three frames of hdr: the value is 0, 1 or 2; 0 select short frame for white balance statistics; 1 select middle frame for white balance statistics;< br/> 2 Select long frames for white balance statistics;

luma limit

Corresponding to the limitRange node in XML, enter the pixel value range of white point statistics, and the points that exceed the range will not be counted.

Name	Description
Y	Y channel value range, the value range is 0~255
R	R channel value range, the value range is 0~255
G	G channel value range, the value range is 0~255
B	B channel value range, the value range is 0~255

mainWindow

Corresponding to the measureWindow node in XML, corresponding to the awb statistics main window configuration

Name	Description
mode	Value 0 or 1 0 Automatically configure the main window of statistics as raw size, the default value is 1 Customize the size of the statistics window
resAll	Enable when mode is 1 Support different measureWindowSize configuration under different resolutions measureWindowSize =[h_offset,v_offset,h_size,v_size], h represents the horizontal direction, v represents the vertical direction

multiwindow

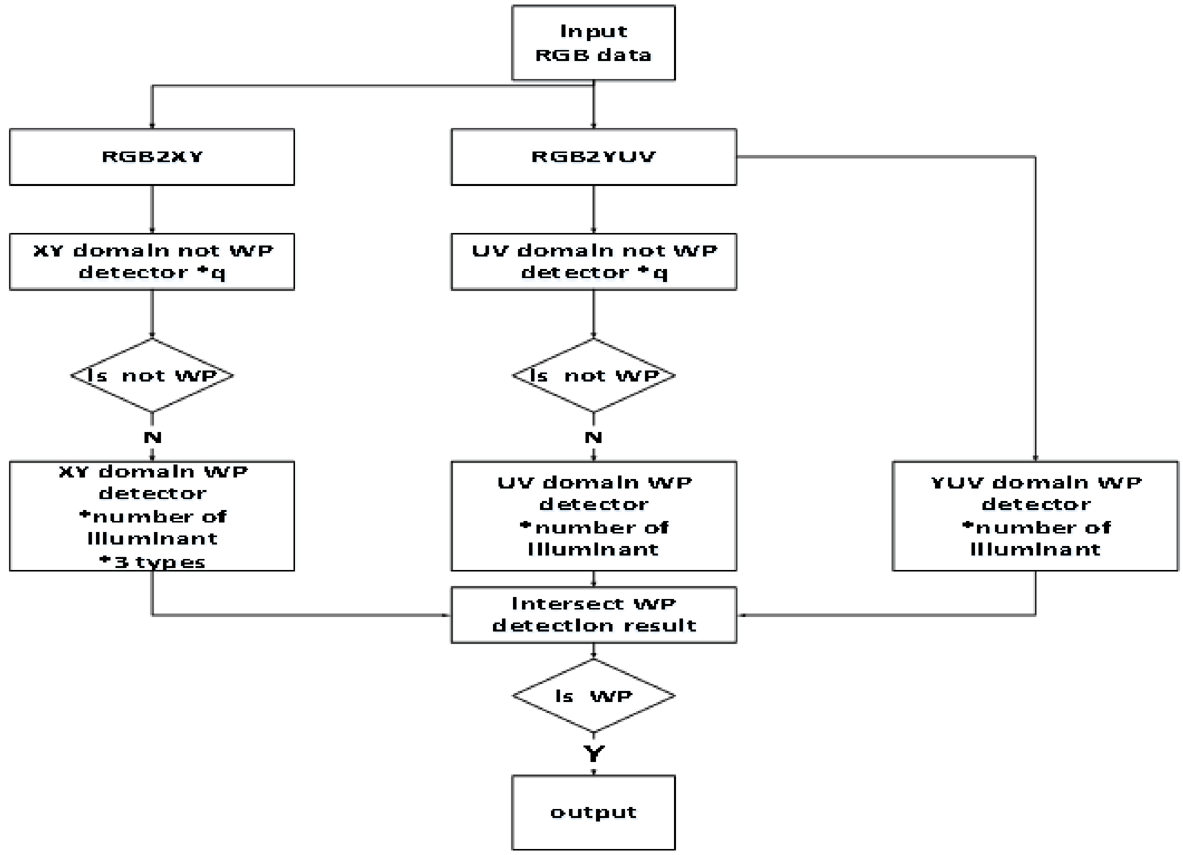
Corresponding to multiWindowEnable and multiwindow in XML, corresponding to multiple sub-window configuration of awb statistics

Name	Description
multiWindowEnable	Value 0 or 1 0 Multiple sub-window statistics disabled 1 Multiple sub-window statistics enabled, default value
multiwindow	Up to 8 sub-windows can be configured Each window can be configured [h_offset,v_offset,h_size,v_size] Value range 0-4095

The statistics of multiple sub-windows need to be combined with the strategy parameters (corresponding to the multiwindowMode in XML). For example, when the face detection module is used, the mode can be set to 1, and the position of the face window can be configured in the multiwindow to achieve the statistical white point minus the skin color point The purpose is to improve the accuracy of white point detection.

Name	Description
multiwindowMode	Multi-window mode The value is 0, or 1 0 means that the statistics of multiple sub-windows are not used. 1 means that the statistics of the white point = the statistics of the main window-the sub-window Statistics

Hardware white point detection process



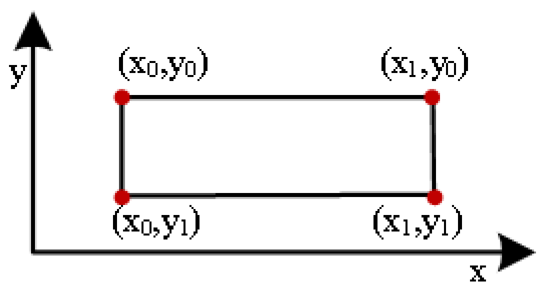
AWB white point detection process

RGB2XY

RGB domain to XY domain transformation parameters, the parameters in XML are as follows

Name	Description
pseudoLumWeight	Make the white points of different light sources lie in a straight line as much as possible, the parameters are generated by the calibration tool, and the value range is 0~1
rotationMat	Rotation matrix, so that the x-axis represents the change of black body radiation color temperature, and the y-axis represents the light source with the same temperature and different spectrum. The parameters are generated by the calibration tool, and the value range is [-3.99,3.99]

XY domain white points detector

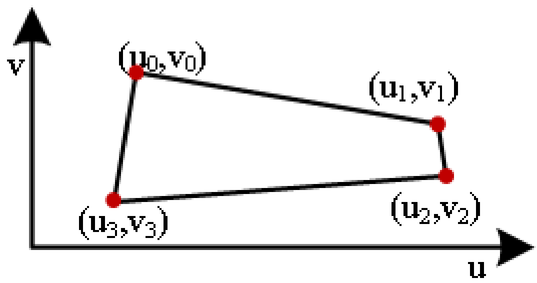


XY domain white point interval

The white point interval of the XY domain is shown above, and the white point in the rectangular frame is the white point. The parameters in the XML are as follows

Name	Description
lightXYRegion	XY domain white point interval, there are three sizes of white point interval, the white point interval is as shown in the figure [x0,x1,y0,y1] Obtained by the calibration tool
normal	Middle frame white point interval, value range [-8,7.99]
big	Big box white point interval, value range [-8,7.99]
small	Small box white point interval, value range [-8,7.99]

UV domain white points detector



UV domain white point interval

The white point interval of the UV domain is shown above, the white point in the quadrilateral box, the parameters in the XML are as follows

Name	Description
lightURegion	The U coordinate of the white point condition in the UV domain forms a closed loop, such as [u0,u1 u2,u3,u0], the value range is [0,255], the decimal place value can only be 0 or 0.5 by the calibration tool Get
lightVRegion	The V coordinate of the white point condition in the UV domain, forming a closed loop, such as [v0,v1 v2,v3,v0], the value range is [0,255], the decimal place value can only be 0 or 0.5 by calibration Tools get

YUV domain white points detector

The corresponding three straight lines are:

(1) Calibrate the uv straight line of the gray block of a certain light source.

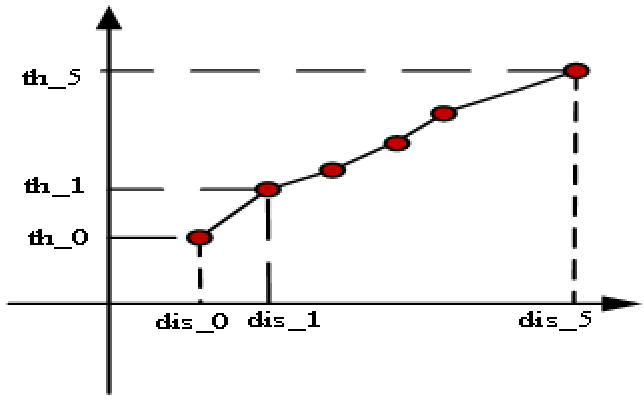
With six gray blocks, under the condition of A light, the uv value can be fitted to a straight line:

$$v = k_0 * u + b_0$$

(2) Calibrate the straight line between the uv value of the gray block and the uv value of the standard white block, dis and y. To require the relationship between the YUV of the gray block, you can first calculate the Euclidean distance between the gray block (u, v) and the standard white point (u_ref, v_ref), and then fit the distance and y into a straight line:

$$y = k_1 * dis + b_1$$

(3) Segmented straight line dis-th



For points in the scene (y0, u0, v0), if $|y0 - y'| < th$, the point is considered to be a white point, otherwise it is not a white point.

The parameters in the corresponding XML are as follows

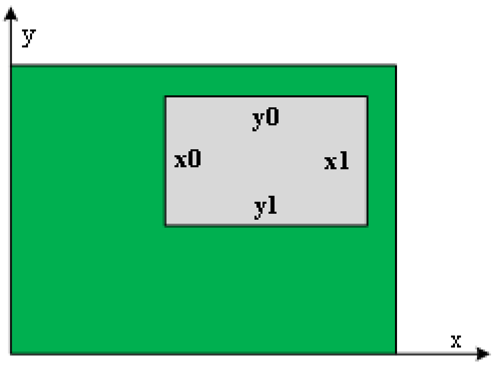
Name	Description
yuvRegion	YUV domain white point interval, obtained by calibration tool
yuvRegion	YUV domain white point interval, obtained by calibration tool
k2Set	k2=-1/k0,k0 is the slope in img. Value range [-2^{18,2}18-1], 9bit signed integer+ 10bit decimal
b0Set	b0 is the intercept in img, value range [0,2^17-1], 17bit unsigned integer + 0bit decimal
k3Set	k3=1/(-1/k0+k0), k0 is the slope in img, Value range [-2^{14,2}14-1], 1bit signed integer + 14bit decimal
k_ydisSet	Straight line! img slope, value range [-2^{18,2}18-1], 9bit signed integer +10bit decimal
b_ydisSet	Straight line! img intercept, value range [-2^{18,2}18-1], 17bit signed integer + 0bit decimal
uRefSet	U value of standard white point. Value range [0,255]
vRefSet	V value of standard white point. Value range [0,255]
disSet	The dis-th of the segmented straight line dis-th. The value range [0,2^12-1], 8bit unsigned integer + 4bit decimal
tHSet	The th of the segmented straight line dis-th, whether it is the white point threshold condition. The value range is [0,2^8-1], 8bit unsigned integer + 0bit decimal

Increase the interval of non-white point, or the interval of white point of additional light source

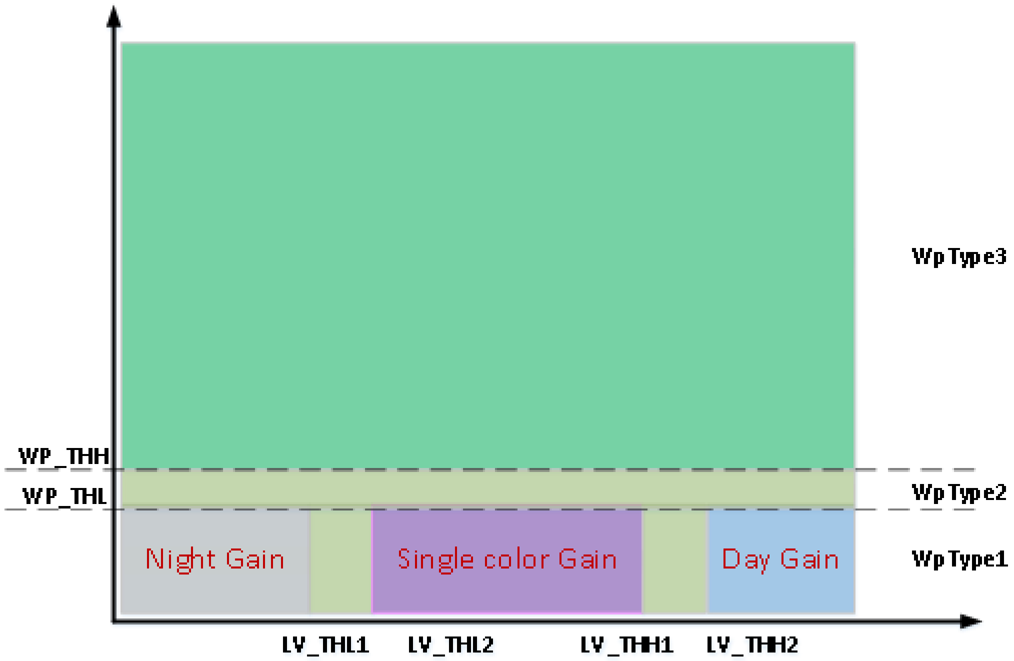
Generally, points that fall within the white point interval of XY, UV, and YUV at the same time will be white points, and some non-white points may also meet this situation, and they are located in the center of the interval, which is difficult to exclude. In this case, a non-white point interval can be added to the UV or XY space, as long as the points fall into this interval, they will be regarded as non-white points.

The points of some light sources deviate far from common standard light sources. If the points of the light source are forcibly added, the white point interval will contain many color blocks. At this time, the white point interval of the additional light source can be added to the UV or XY space. As long as the white point falls within this interval, it will be regarded as a white point.

The maximum sum of the number of non-white point intervals and white point intervals of additional light sources is 7. Corresponding to the excludeRange parameter in XML,



Name	Description
Domain	Value is 0 or 1 0 UV domain white point interval 1 XY domain white point interval
mode	Value 1 or 3 1 The range is the non-white point interval 3 The range is the white point interval of the additional light source
window	The configuration interval is shown in the figure above [x0,x1,y0,y1] When Domain=0, the value range is [0,511], where 1bit is a decimal place When Domain=1 , The value range is [-8192,8191], where 10bit is the decimal place



AWB partition strategy calculation WBGain schematic

(1) Interval parameters

According to the lower threshold, the environment brightness-white point number space is partitioned, so that different sections use different methods to calculate the white balance gain. The corresponding parameters in the XML are as follows

Name	Description
LV_THL	The ambient brightness threshold on the picture LV_THL1 The value range is 0-xxx
LV_THL2	The ambient brightness threshold on the picture LV_THL2 The value range is 0-xxx
LV_THH	The ambient brightness threshold on the picture LV_THH1 The value range is 0-xxx
LV_THH2	The ambient brightness threshold on the picture LV_THH2 The value range is 0-xxx
WP_THL	The threshold of the number of white dots on the graph WP_THL The value range is 0-xxx The actual number of white dots is compared with WP_THL*totalPixe*100000, where totalPixel = wight * height / ds_w/ds_h, the image width and height are wight, height, and the horizontal and vertical downsampling multiples are ds_w, ds_h
WP_THH	The threshold of the number of white dots on the graph WP_THH The value range is 0-xxx The actual number of white dots is compared with WP_THH*totalPixel*100000

The parameters in WPType3 are mainly determined by white point statistics. The parameter in WPType1 is wbgain. If it is the first frame, it may be a fixed Night Gain or DayGain, or it may be WBGain calculated by a simple color algorithm. WPType2 is the transition zone. Corresponding to the parameters in XML,

① DayGain

Name	Description
spatialGain_L	Recommended wbgain under particularly bright sunlight, value range [0.5-3.9]
spatialGain_H	Recommended wbgain under normal sunlight, value range [0.5-3.9]

② NightGain

Name	Description
temporalDefaultGain	Recommended wbgain when the ambient brightness is low, value range [0.5-3.9]

③ SingleColorGain (singleColorProcess)

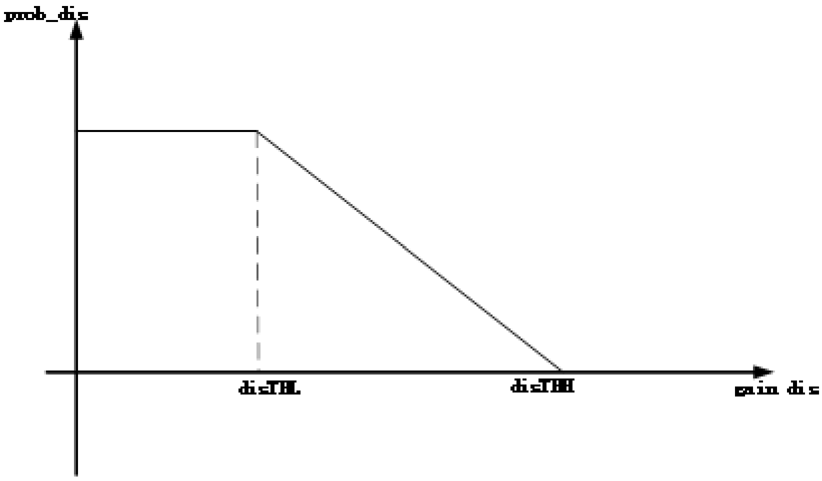
Name	Description
colorBlock	Select the color block used for pure white balance. You can choose from the color card or define it yourself. Generated by calibration tool
index	Color index, representing color Value range 0-xxx The calibration tool defaults to 13, 14, 15, 16, 5, 10 on the x-rite color card
meanC	The average chromaticity value of the color in the LCH space
meanH	The average hue value of the color in the LCH space
lsUsedForEstimation	Select the light source for simple color, generated by the tool
name	Light source name
RGain	The white balance gain of the red channel of the light source The value is greater than 0 decimal
BGain	The white balance gain of the blue channel of the light source The value is greater than 0 decimal
alpha	H weight in LCH space The value range is 0.0-1.0

(2) Light source weight parameter

Probability calculation parameters of different light sources in WpType3

$$Prob_i = probLV_i * probDis_i * probWP_i$$

① $probDis_i$ Distance probability parameter



The parameters in the corresponding XML are as follows

Name	Description
proDis_THL	The distance threshold disTHL on the graph The value range is 0-xxx
proDis_THH	The distance threshold disTHH on the graph The value range is 0-xxx

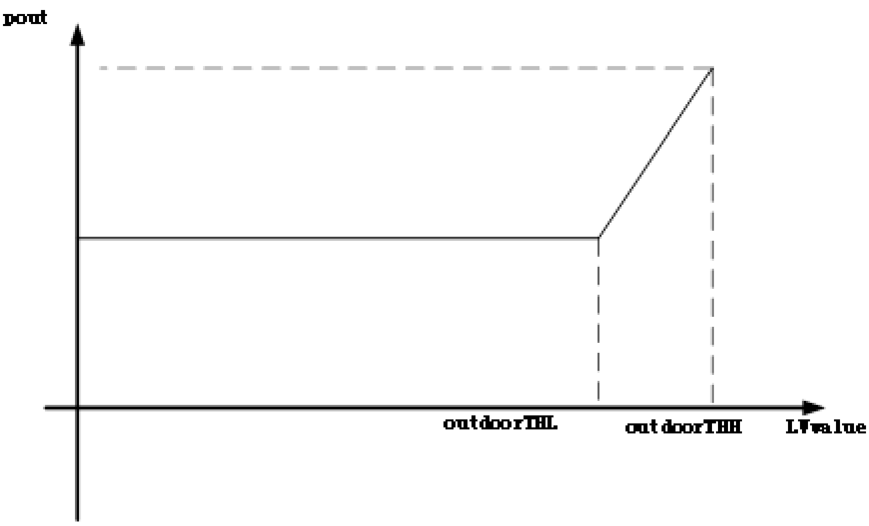
② $probLV_i$

Scene brightness probability parameter

For outdoor light sources, the probability of the light source is pout, which is calculated according to the brightness-pout curve in Figure 4- 14

For indoor light sources, the probability of the light source is pin = 1-pout

pd50 = max(pout, pin)



The parameters in the corresponding XML are as follows

Name	Description
proLV_Outdoor_THL	The ambient brightness threshold outdoorTHL on the map The value range is 0-xxx
proLV_Outdoor_THH	The ambient brightness threshold outdoorTHH on the map The value range is 0-xxx

③ $probWP_i$

Probability parameter of the number of scene white points

The parameters in the corresponding XML are as follows

Name	Description
wpNumPercTh	For light sources whose number of white points is less than wpNumPercTh*totalPixel, the probability of the number of white points is 0, where totalPixel = wight* height /ds_w/ds_h, the image width and height are wight, height, and the downsampling multiples in the horizontal and vertical directions are respectively ds_w, ds_h The value range is 0-xxx

(3) White point weight parameter

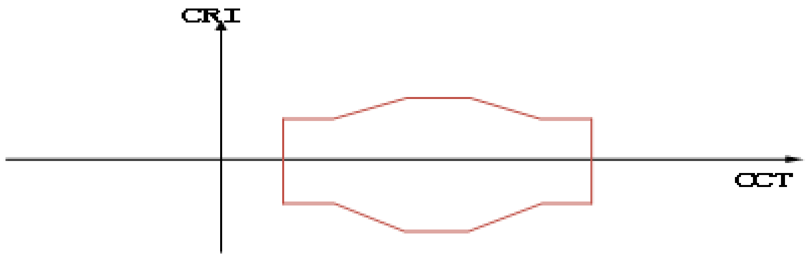
Name	Description
staWeigthSet	Corresponding to different weights of wbgain calculated from the white point under different brightness, corresponding to LvMatrix in XML. The value is 0-100 The weight of DayGain is 100-staWeigt

WBGain color adaptation adjustment

The corresponding parameters in XML are as follows,

Name	Description
ca_Enable	The white balance corrected image is as consistent as possible with the color appearance perceived by the human eye. The enable value is 0 or 1, which means disable and enable, respectively.
ca_LACalcFactor	The factor that controls the degree of color adaptation under different brightness, the default value is 40.
ca_TargetGain	The factor that controls the degree of color adaptation under different brightness, the default value is 40. The value range is 0.0-xxx, and the default value is d50 white balance gain.

WBGain range limit



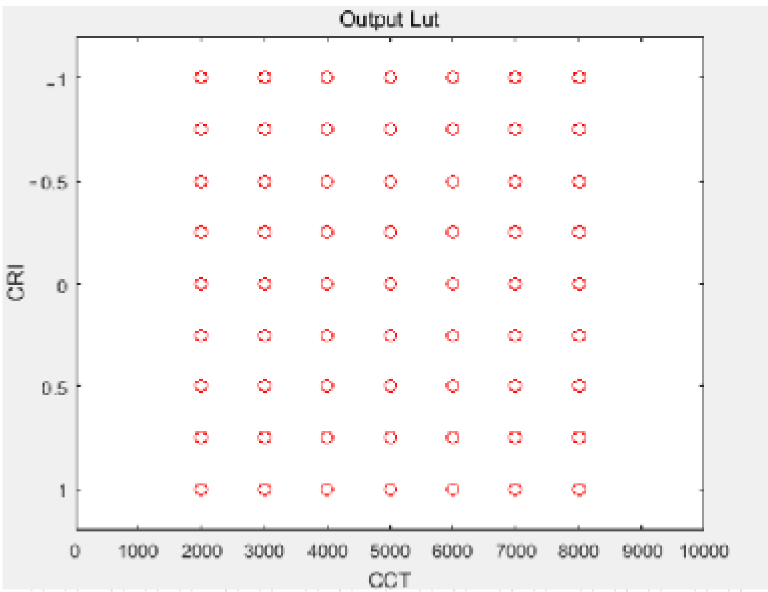
Limit the white balance gain to the area surrounded by the red straight line as shown in the figure above. The parameters in the XML are as follows

Name	Description
wbGainDaylightClipEnable	Outdoor minimum color temperature limit enable The value is 0 or 1, which respectively represent disable and enable
wbGainClipEnable	Color temperature range limit enable The value is 0 or 1, which respectively represent disable and enable
wbGainDaylightClip	Outdoor minimum color temperature limit parameter
outdoor_cct_min	The minimum outdoor color temperature is not limited
wbGainClip	Color temperature range limit
cct	The number of color temperature sampling points, the value is 0-7
cri_bound_up	The upper boundary value of cri corresponding to cct The value is not limited
cri_bound_low	The lower boundary value of cri corresponding to cct The value is not limited

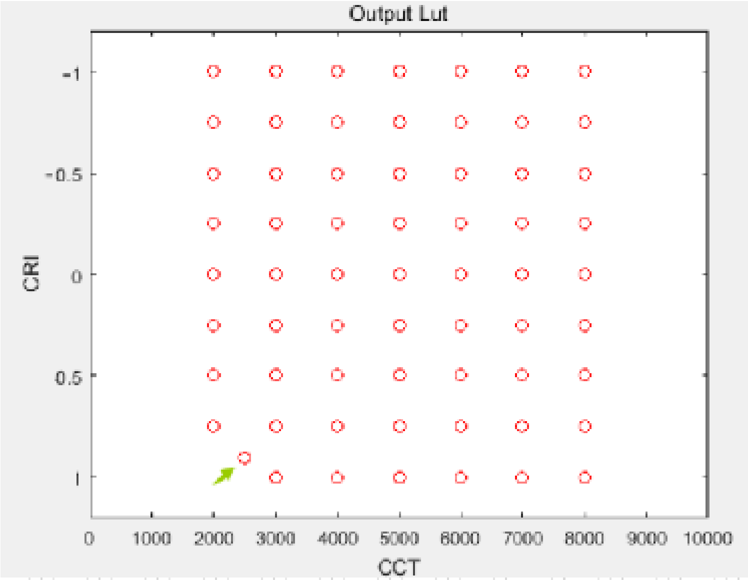
WBGain tone adjustment

Name	Description
wbGainAdjustEnable	Hue adjustment enable The value is 0 or 1, which means disable and enable respectively
ct_grid_num	Enter the number of sampling points of the color temperature of the color temperature table value 0-10
ct_in_range	Enter the color temperature range of the color temperature table The value is not limited
cri_grid_num	Enter the number of sampling points of the color rendering index of the color temperature table value 0-10
cri_in_range	Enter the color rendering index range of the color temperature table The value is not limited
lutAll	Different ambient brightness can be configured with different output color temperature meters, and up to three ambient brightness can be configured
LvValue	Ambient brightness Value range 0-xxx
ct_out	The ct value of each point is shown in the following table, from left to right (the hue is from cold to warm), from top to bottom The value range is 0-xxxx
cri_out	The cri of each point as shown in the following table, from left to right, from top to bottom (hue from purple to green) The value range is not limited

Output color temperature-color rendering index table 1 (without color temperature adjustment)



Output color temperature-color rendering index table 1 (partial color temperature adjustment)



2.3 Calibration

Basic principles of AWB calibration

Mainly to calibrate the white point conditions of Raw in XY, UV, YUV, pure color algorithm parameters and white balance gain under standard light source

Raw image requirements for AWB calibration

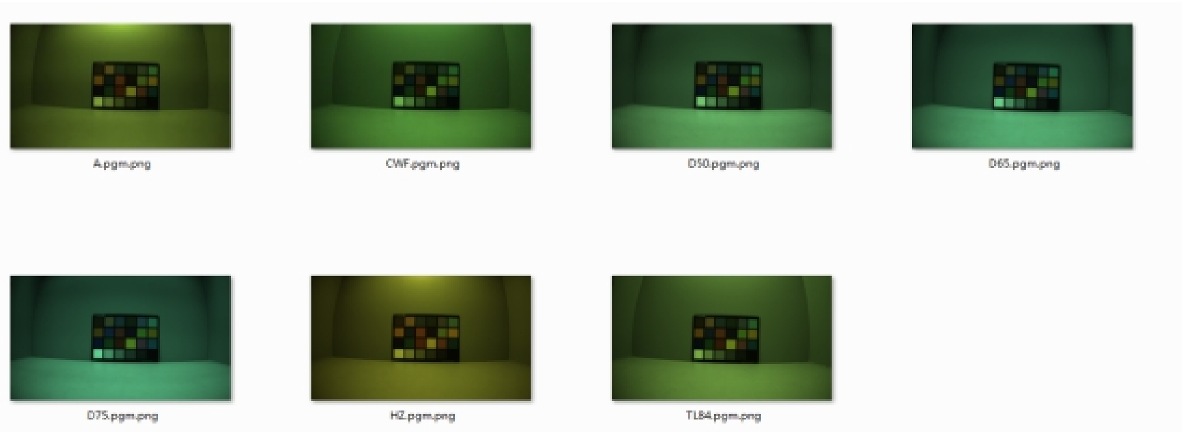
When collecting raw images, you need to prepare the following environment:

① Equipment: x-rite 24 color card, light box

② Adjust the exposure parameters so that the maximum value of the brightest white block in the color card is [150-240], the brighter the better in this range

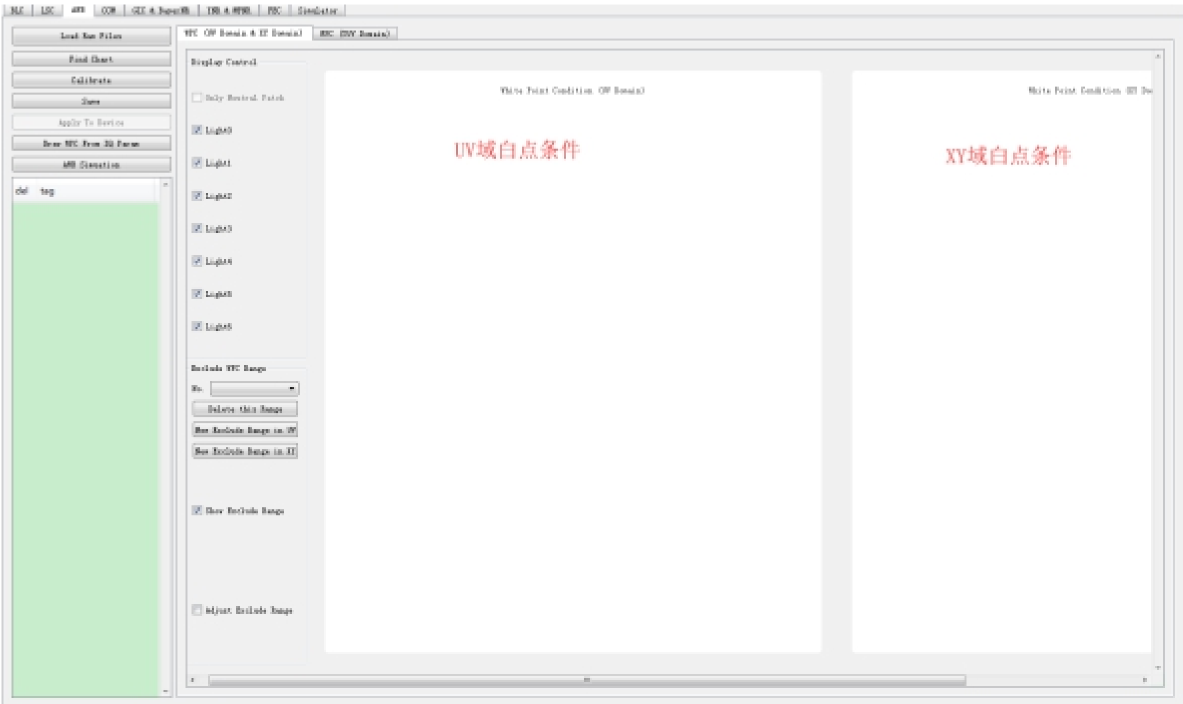
③ The color card occupies more than 1/9 of the screen

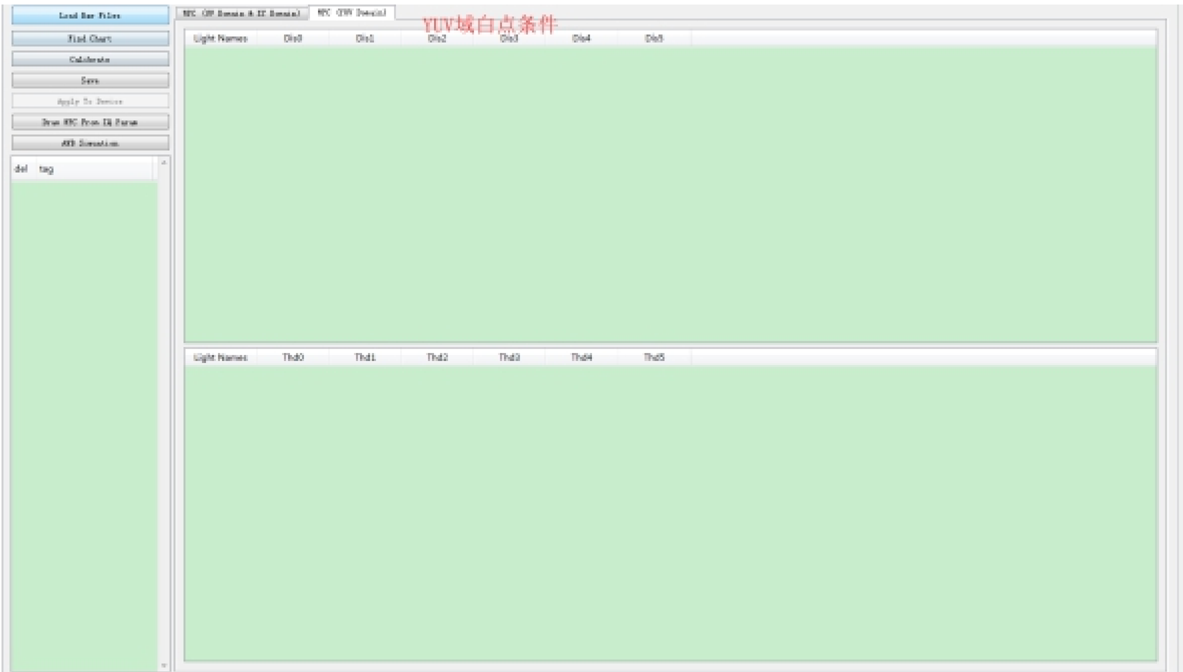
Take the x-rite 24 color card under A, CWF, D50, D65, D75, HZ, TL84 light source in sequence, and the schematic diagram of the mosaic is as follows:



AWB calibration tool interface description

(1) When calibrating, it is mainly to adjust the white point boundary of the UV and XY domains, and the TH value of the YUV domain

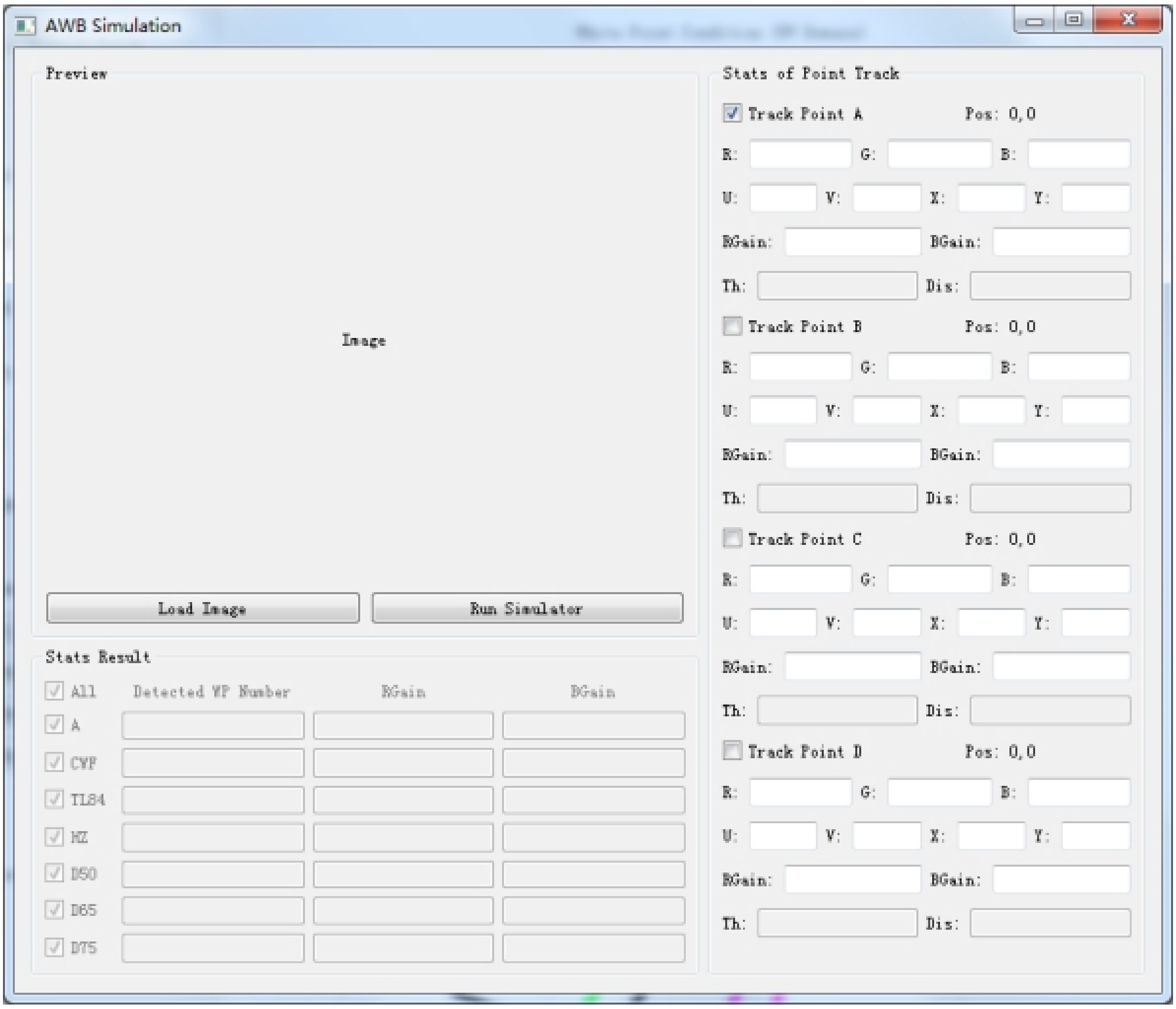




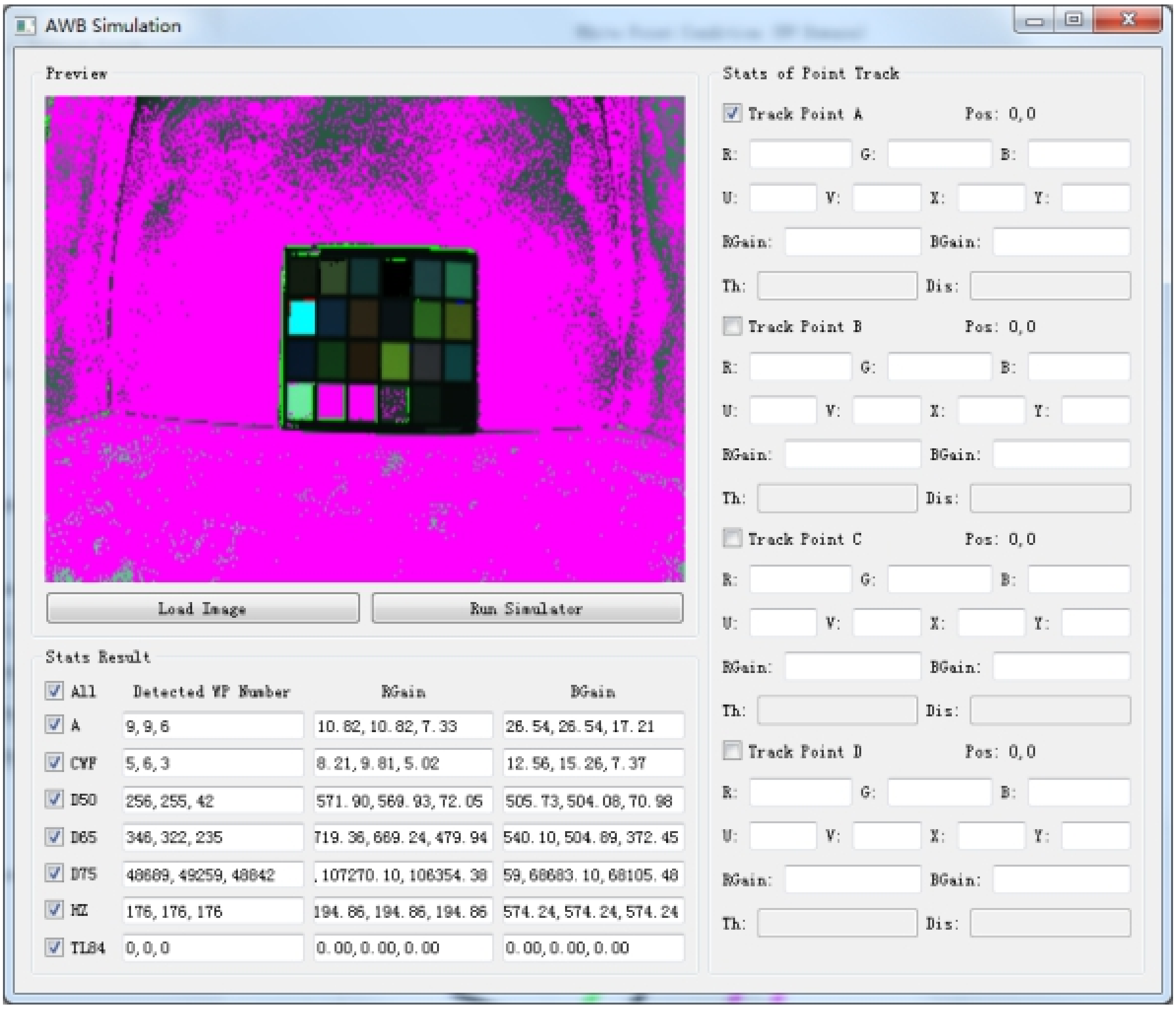
(2) The information display of each light source can be selected through the check box in front of LightX in the Display Control panel.

(3) The Exclude WPC Range panel can be used to increase the non-white point interval and the additional light source white point interval.

(4) AWB Simulaton is used to detect the white point of the raw image and calculate the white point gain



① LoadImage After importing the Raw image, the white point information will be printed as shown below. The white points of different light sources are displayed in different colors. The number of white points in the middle, large, and small boxes RGain accumulation and BGain accumulation will be displayed in the three text boxes of Detected WP Number, RGain, and BGain



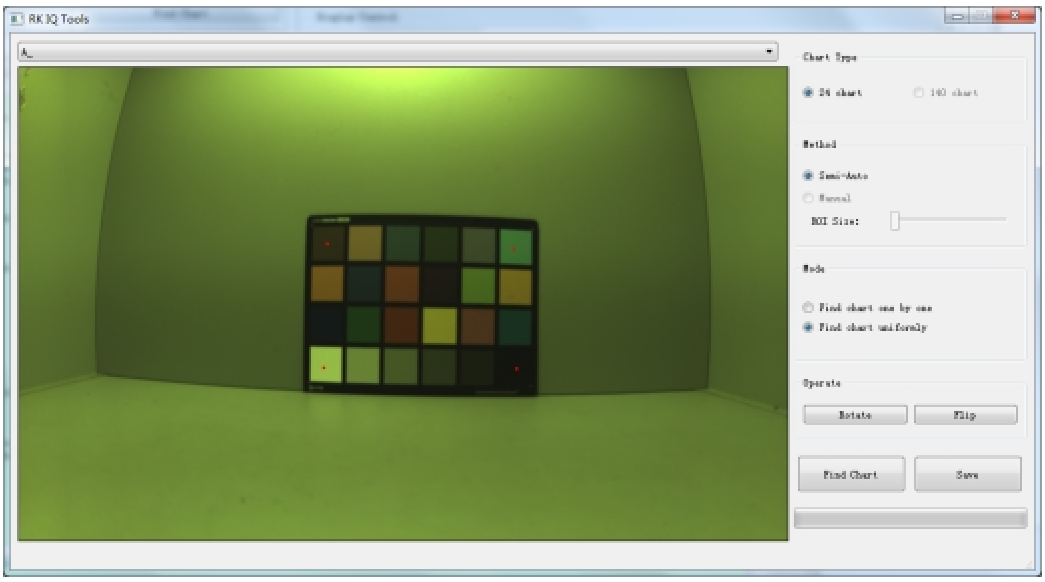
② Click any position in the image, it will be mapped to the UV domain white point condition interface and the XY domain condition interface, which is convenient to check whether the point falls within the white point interval, and the RGBUVXY RGain BGain Dis Th of the point will be displayed on this interface On the Stats of Point Track panel

AWB calibration steps

(1) When AWB is calibrated, BLC and LSC must be calibrated

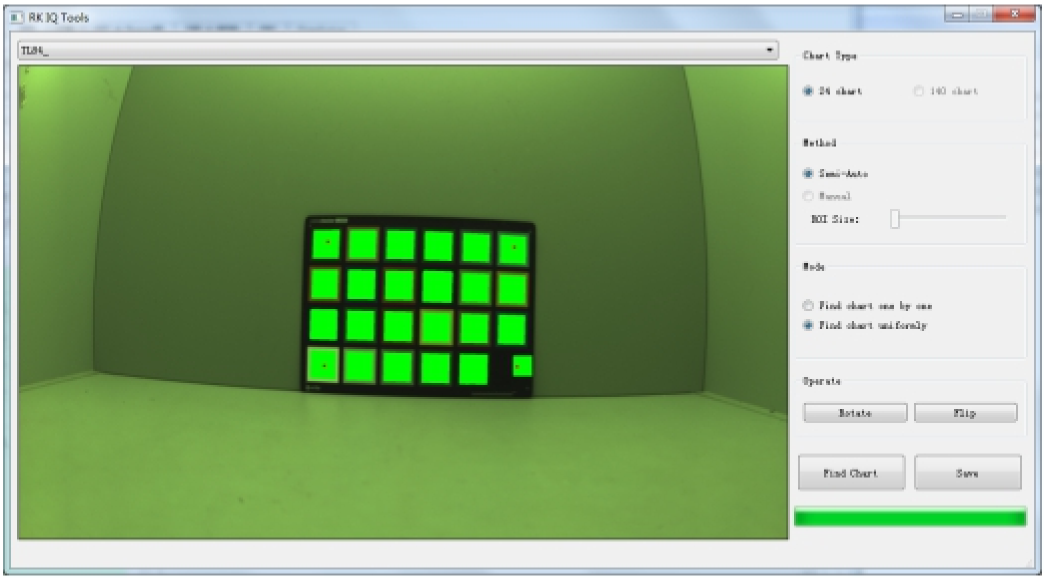
(2) Click Load Raw Files to import the raw images under A, CWF, D50, D65, D75, HZ, TL84 (it is recommended to calibrate the raw images of these seven light sources)

(3) Click Find Chart to identify the color chart

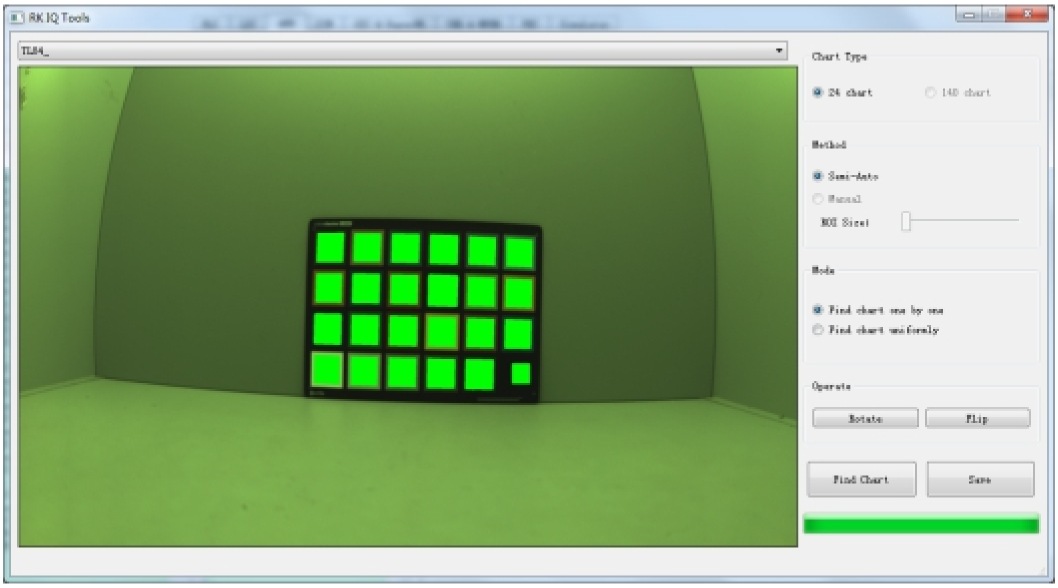


① Click block 1, block 6, block 19, block 20 in turn

② Click FindChart to identify the color patches of all light sources in batches, as shown below (display the white point detection result of the last light source)

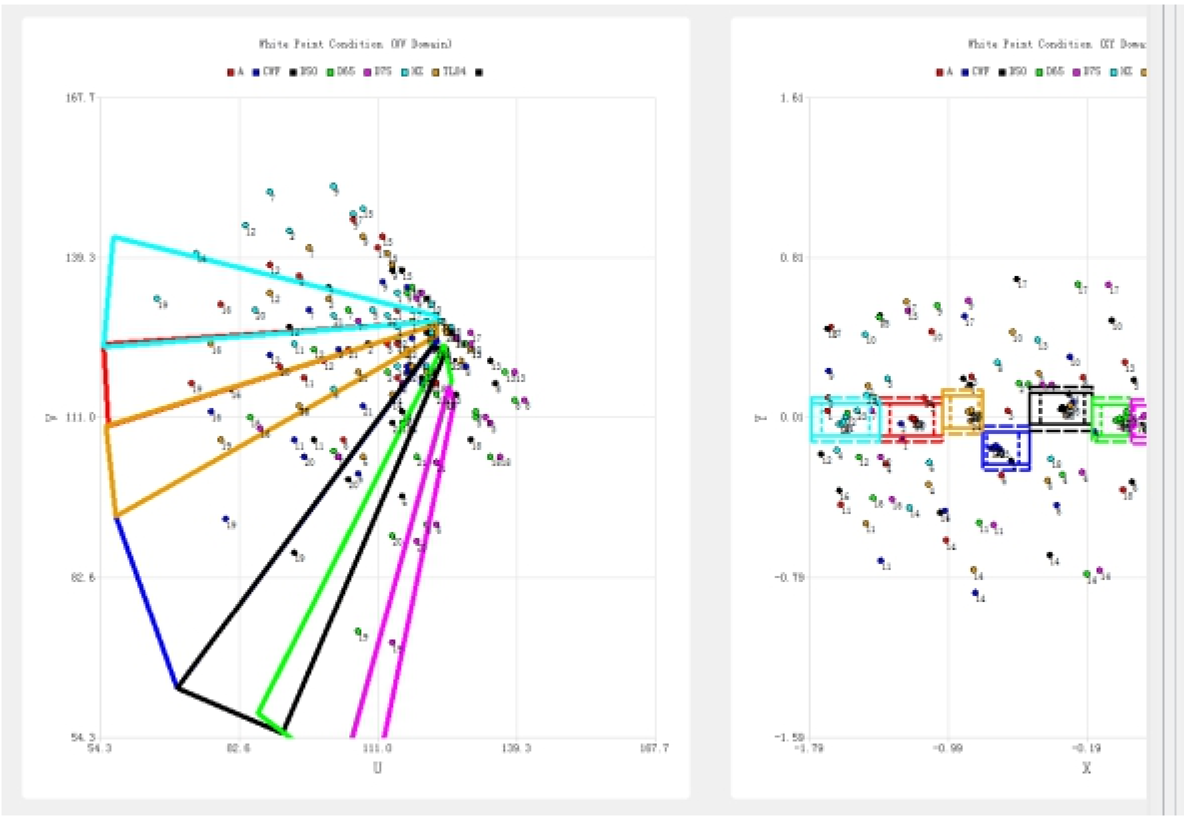


③ Select other light sources from the drop-down menu, confirm the correctness of the color block recognition, and found that only the last block of TL84 is recognized a bit to the right. At this time, you only need to re-check it separately. Select Find chart one by one in the solid mode and repeat step 12. Until the color block of the TL84 color card is recognized correctly, as shown below



④ Click Save to complete recognition

(4) Click Calibrate to get the following initial white point conditions and other parameters



WPC (UV Domain & XY Domain) | WPC (YUV Domain)

	Light Names	Dis0	Dis1	Dis2	Dis3	Dis4	Dis5
1	A	44	108	236	364	620	876
2	CWF	39	103	231	359	615	871
3	D50	30	94	158	414	542	798
4	D65	18	82	210	338	594	850
5	D75	7	71	199	327	583	839
6	HZ	50	114	242	370	626	882
7	TL84	38	102	166	294	550	806

	Light Names	Thd0	Thd1	Thd2	Thd3	Thd4	Thd5
1	A	11	14	17	20	23	26
2	CWF	11	14	17	20	23	26
3	D50	11	14	17	20	30	40
4	D65	11	14	17	20	23	26
5	D75	11	14	17	20	23	26
6	HZ	11	14	17	20	23	26
7	TL84	11	14	17	20	23	26

(5) Click AWB Simulaton, and then import raw images under A, CWF, D50, D65, D75, HZ, TL84 to check the accuracy of white point detection

(6) Modify the UV domain or XY domain frame or YUV TH to make the white point detection of the color card under each light source more accurate

(7) Click Save

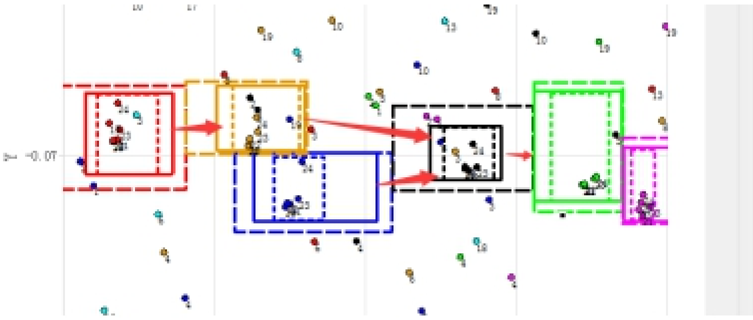
(8) Repeat (5) ~ (7) until the white point detection of each light source is reasonable.

(9) Matters needing attention:

① Adjust the border as far as possible so that the white point is inside the frame, and the non-white point is outside the frame (generally not possible)

② The interval enclosed by the middle or large frames of all light sources must be closely connected (three line types represent three size frames)

Error demonstration (the intervals of the big boxes are closely connected, but there are intervals between the middle boxes, as shown by the arrows below):



Correct demonstration:

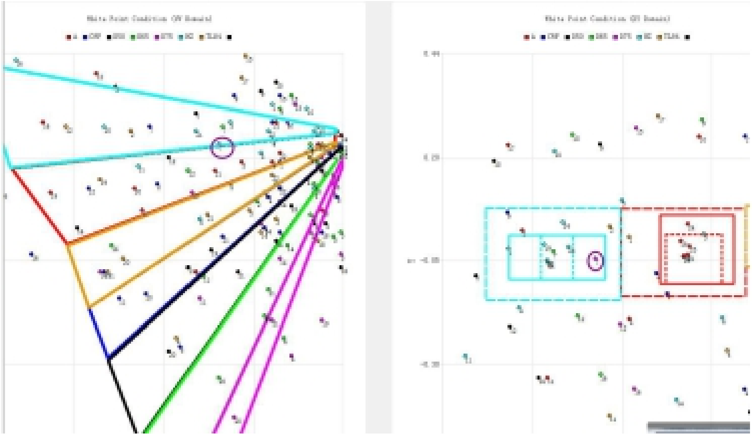
③ a and hz light sources can be more compact in the Y direction of the XY domain, and d50 d65 can be relaxed in the Y direction of the XY domain

④ The interval enclosed by all light sources in the UV domain must be closely connected

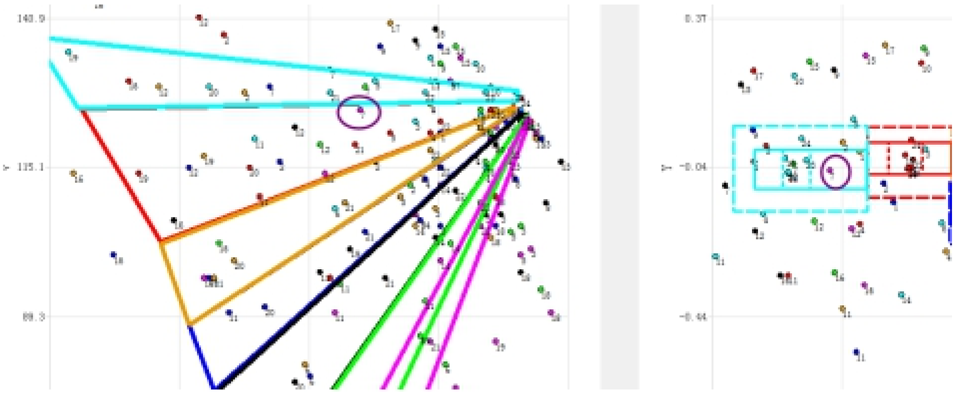
⑤ Different light source boundaries can overlap, but do not overlap in both XY and UV space at the same time

⑥ Refer to the XY space to divide the UV space to exclude non-white points

If the 7th block of the circled D75 light source falls within the hz range, it will be recognized as a white spot



After re-adjustment, the 7th block B of D75 light source is not in the same light source in xy and uv space, and will not be recognized as a white point

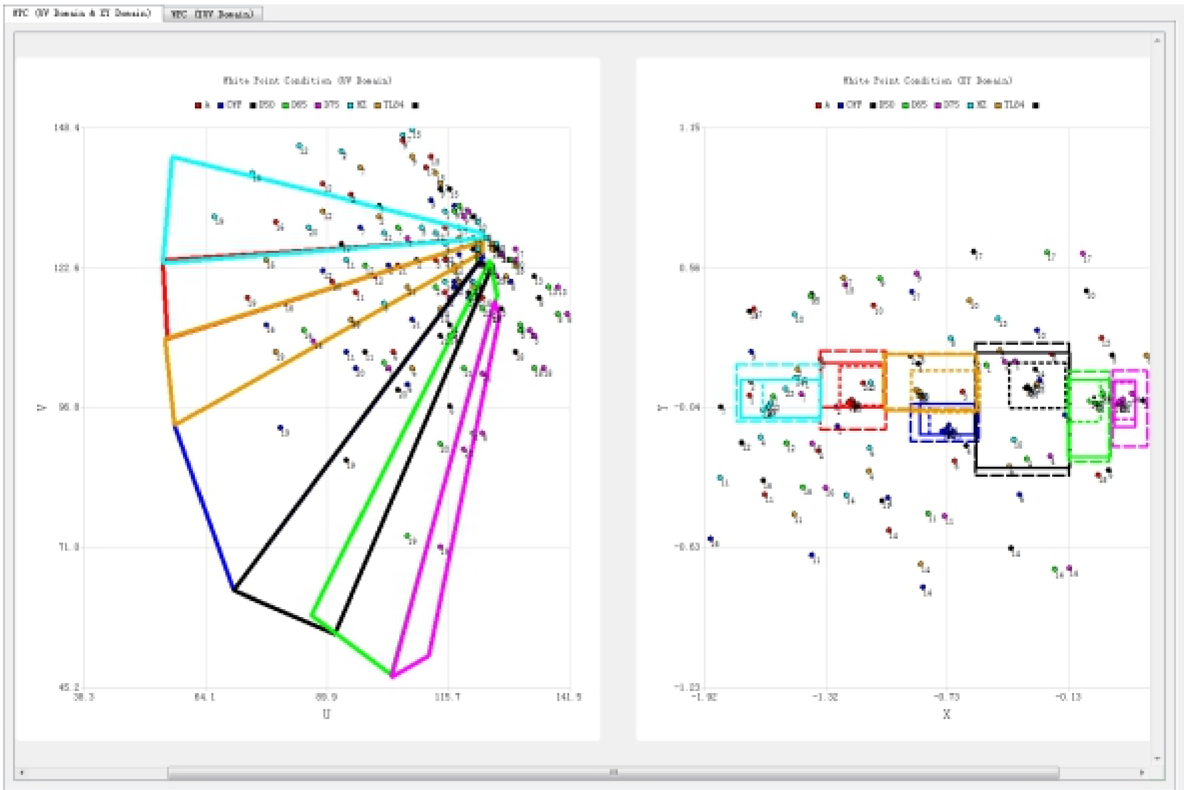


⑦ When the non-white point falls in the white point interval of XY and UV, it can also be eliminated by reducing the TH or increasing the non-white point interval.

⑧ When the white point falls in the white point range of XY and UV, but it is still not a white point, it may be excluded because it exceeds the brightness range, or it falls in the non-white point range, or it does not fall in the range because it is less than TH. In the white point interval of the YUV domain

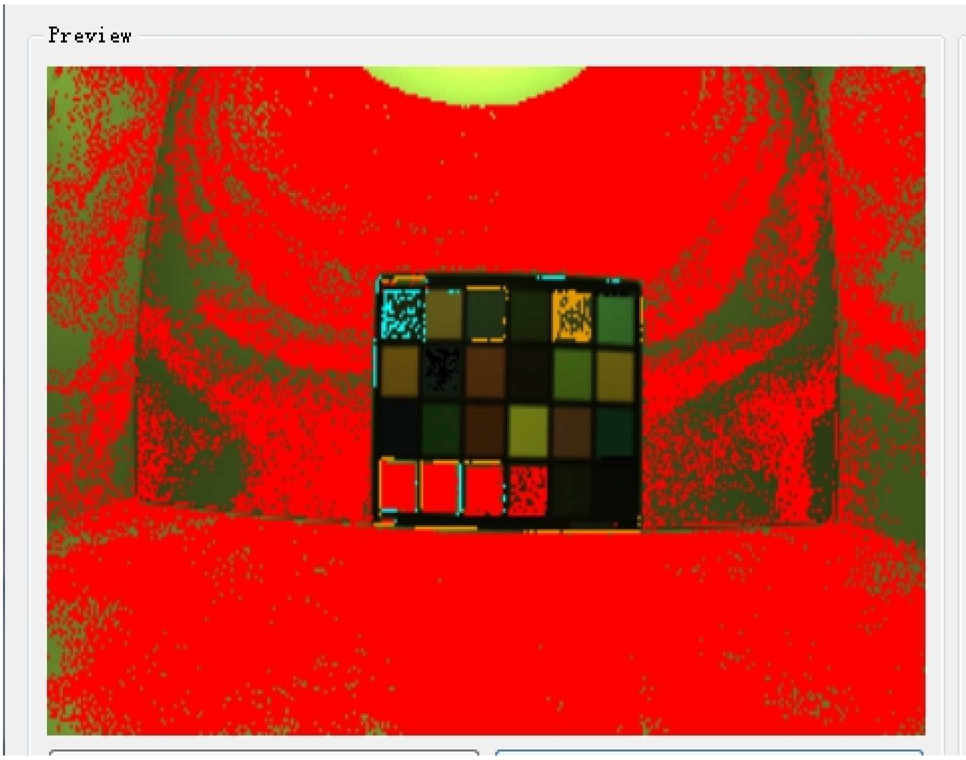
AWB calibration result

Final white point conditions:

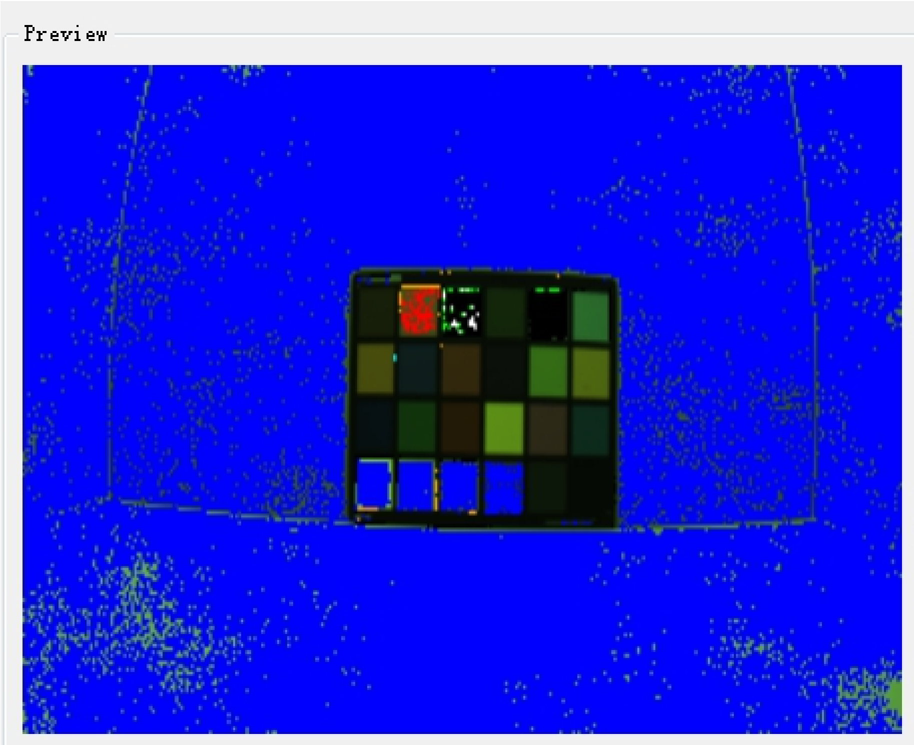


The white point detection result is:

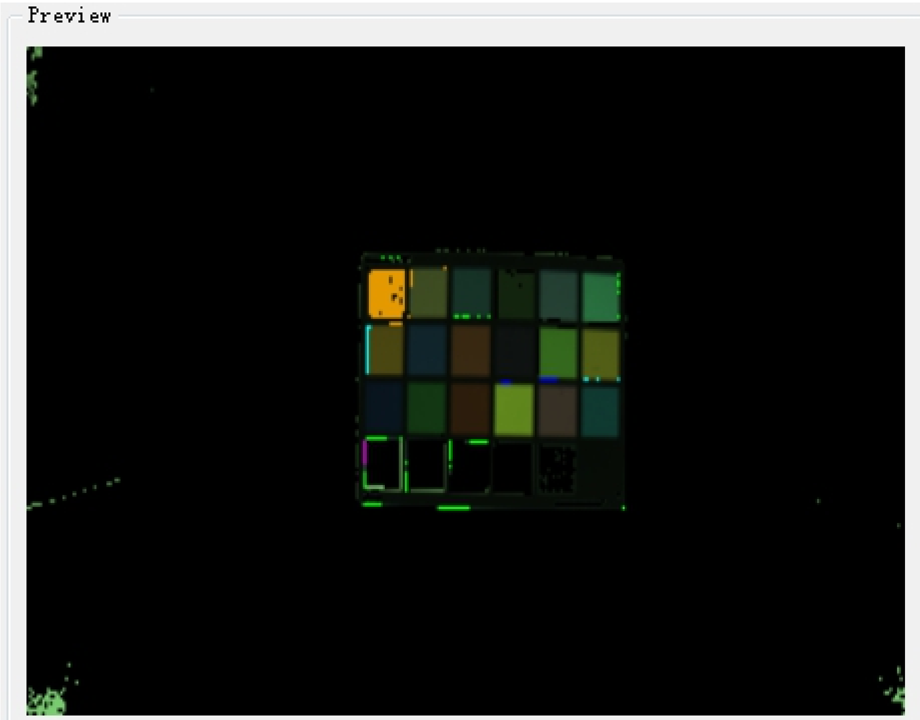
A light



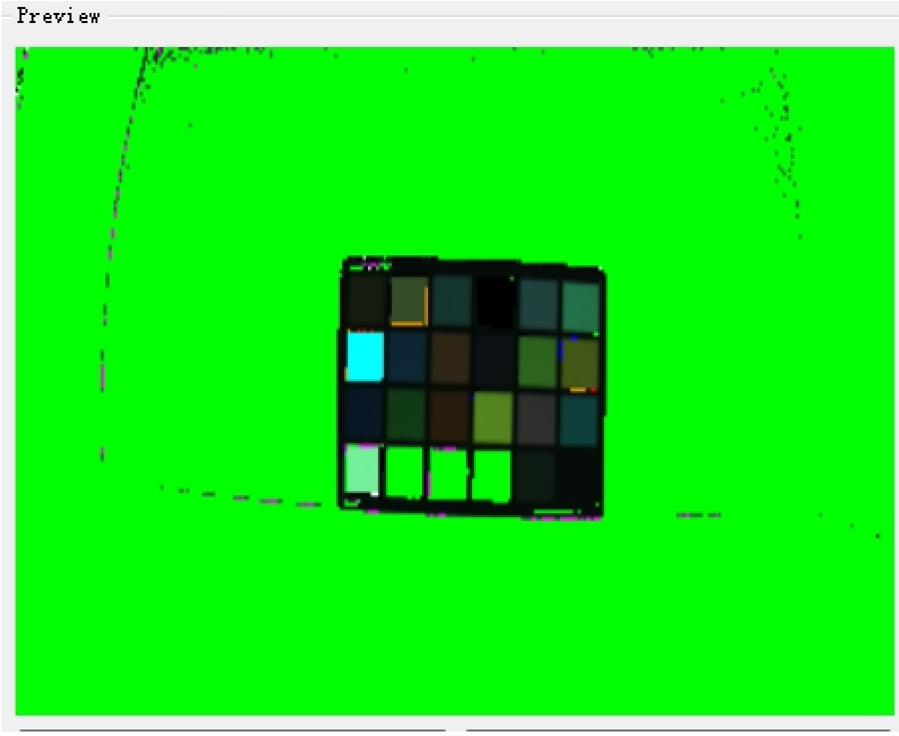
CWF:



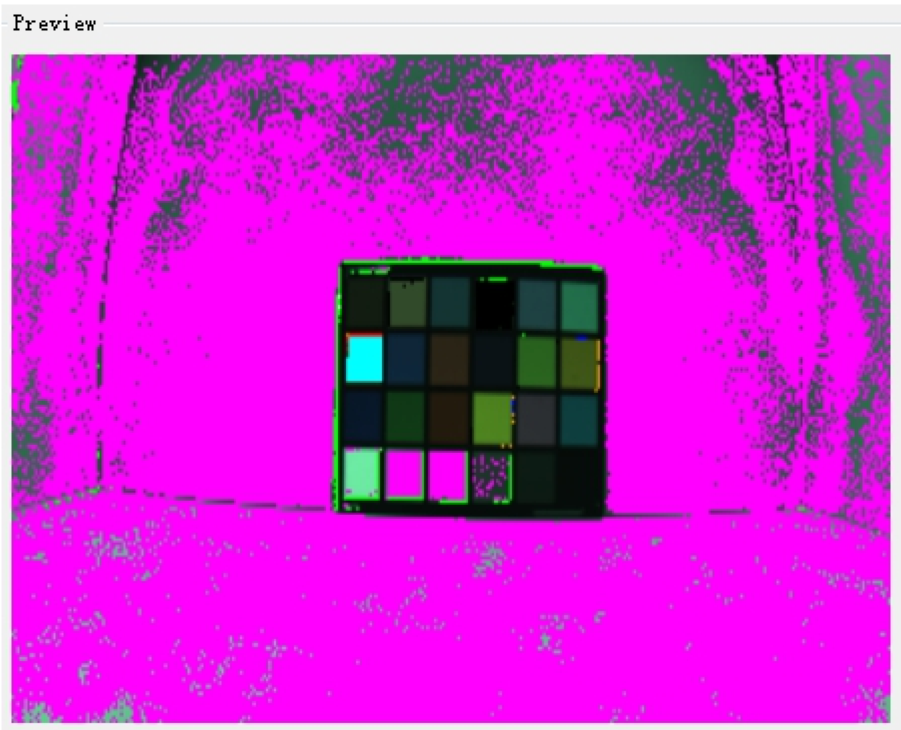
D50



D65



D75



HZ

TL84

2.4 Common problem positioning

```
In order to solve the problem of abnormal white balance, it is usually necessary
to capture the log and capture the raw to analyze the cause, and solve it by
modifying the white point condition or modifying the policy parameter.
```

Grab log and analyze

The awb log level used for debugging is export persist_camera_engine_log=0x2ff4

AWB log interpretation

(1) Control and mode log

① AwbReConfigV200: byPass: 0 mode:1

byPass is 0 means white balance correction is enabled, 1 means white balance correction is disabled

mode is 0 means that the current is manual white balance mode, is 1 means that the current is automatic white balance mode

(2) Algorithm related log

```
******************************326th frame*************************
awbConverged(1) frameChoose(1) ,LVValue(16460),LVLevel(11),LVType(3)
WPNo(15887,20071,21455),effective xy type (0),effective uv type(255,0),vaild wp number(20071),WPmode(3),WPType(3)
current gain (rggb):(1.521681,1.000000,1.000000,2.308375) spaGainEqu2Tem(0) clip(0) df(0.90)
wbGainSgc (rggb):(0.000000,0.000000,0.000000,0.000000) ,wbWeightSgc(-1.000000),sgcGainEqu2Tem(0)
wbGainSpa (rggb):(0.000000,0.000000,0.000000,0.000000) ,wbWeightSpa(-1.000000)
wbGainTepType1 (rggb):(0.000000,0.000000,0.000000,0.000000)
wbGainType1 (rggb):(0.000000,0.000000,0.000000,0.000000)
wbGainType3(rggb):(1.521560,1.000000,1.000000,2.308260),wbWeightType3(1.000000)
the light source slecetion to 3dyuv is :[0,2,6,1]
: A:
:      stategy_result.gain (rggb):(1.288514,1.000000,1.000000,2.826975)
:      prob_total(0.372926),prob_dis(0.151197),prob_LV(0.142857),prob_WPNO(0.383987)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(1.287584,2.843158) WPNo(7707)
:      type1: gain (rg,bg):(1.291619,2.862351) WPNo(8557)
:      type2: gain (rg,bg):(1.288622,2.825095) WPNo(6905)  Weight(0.895939)
: CWF:
:      stategy_result.gain (rggb):(1.652013,1.000000,1.000000,2.347027)
:      prob_total(0.014538),prob_dis(0.162406),prob_LV(0.142857),prob_WPNO(0.013851)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(1.645590,2.367552) WPNo(278)
:      type1: gain (rg,bg):(1.654612,2.362139) WPNo(349)
:      type2: gain (rg,bg):(1.652648,2.344999) WPNo(253)  Weight(0.910072)
: D50:
:      stategy_result.gain (rggb):(1.784953,1.000000,1.000000,1.763579)
:      prob_total(0.369272),prob_dis(0.155234),prob_LV(0.142857),prob_WPNO(0.370335)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(1.770483,1.780922) WPNo(7433)
:      type1: gain (rg,bg):(1.768410,1.780313) WPNo(7542)
:      type2: gain (rg,bg):(1.790423,1.757024) WPNo(5394)  Weight(0.725683)
: D65:
:      stategy_result.gain (rggb):(1.969202,1.000000,1.000000,1.566281)
:      prob_total(0.003728),prob_dis(0.133899),prob_LV(0.142857),prob_WPNO(0.004335)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(1.979457,1.567466) WPNo(87)
:      type1: gain (rg,bg):(1.960826,1.550404) WPNo(94)
:      type2: gain (rg,bg):(1.968164,1.566162) WPNo(79)  Weight(0.908046)
: D75:
:      stategy_result.gain (rggb):(2.144231,1.000000,1.000000,1.420382)
:      prob_total(0.000119),prob_dis(0.123669),prob_LV(0.142857),prob_WPNO(0.000149)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(2.144231,1.420382) WPNo(3)
:      type1: gain (rg,bg):(2.144231,1.420382) WPNo(3)
:      type2: gain (rg,bg):(2.144231,1.420382) WPNo(3)  Weight(1.000000)
: HZ:
:      stategy_result.gain (rggb):(1.129557,1.000000,1.000000,3.248359)
:      prob_total(0.005423),prob_dis(0.106575),prob_LV(0.142857),prob_WPNO(0.007922)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(1.000000)
:      type0: gain (rg,bg):(1.130459,3.248583) WPNo(159)
:      type1: gain (rg,bg):(1.129604,3.239437) WPNo(165)
:      type2: gain (rg,bg):(1.129540,3.248355) WPNo(156)  Weight(0.981132)
: TL84:
:      stategy_result.gain (rggb):(1.470842,1.000000,1.000000,2.329216)
:      prob_total(0.233994),prob_dis(0.166021),prob_LV(0.142857),prob_WPNO(0.219421)
:      spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126),statistics gain weigth(0.900000)
:      type0: gain (rg,bg):(1.430740,2.439226) WPNo(4404)
:      type1: gain (rg,bg):(1.423992,2.404189) WPNo(4745)
:      type2: gain (rg,bg):(1.444306,2.362594) WPNo(3097)  Weight(0.703224)
: wbGainTepTp3 (rggb):(1.259000,1.000000,1.000000,2.000000)  wbGainTepTp3 (0.000000)
: excWpRangeResult[0]:   gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(0)
: excWpRangeResult[1]:   gain (rg,bg):(1.274361,2.514130) WPNo(2004),mode(1)
: excWpRangeResult[2]:   gain (rg,bg):(1.932277,1.410095) WPNo(23),mode(1)
: excWpRangeResult[3]:   gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(1)
: excWpRangeResult[4]:   gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(0)
: excWpRangeResult[5]:   gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(0)
: excWpRangeResult[6]:   gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(0)
```

① awbConverged is 0 or 1, respectively, AWB has not converged and converged; frameChoose is 0 1 2 respectively indicates that the short, medium, and long frames are used as the input of AWB hardware statistics; LVValue is the ambient brightness value

② WPNo is the number of white points in the small, middle, and large boxes

③ The effective xy type is 0 1 2 which means that the white points of the middle frame, the large frame, and the extra light source are used to calculate WBGain, and the vaild wp number is the corresponding number of white points (the sum of the points greater than wpNumPercTh); WPType is 1 2 3 corresponding to the partition WpType1, WpType2, WpType3 in the strategy map

④ current gain is the wbgain calculated this time; df is the current dampfactor parameter

⑤ wbGainSgc is the WBGain calculated by the simple color algorithm; wbWeightSgc is the weight of wbGainSgc, -1 means that the algorithm is not used

⑥ wbGainSpa is dayGain, wbWeightSpa is the weight of dayGain, -1 means not used

⑦ wbGainType1 is WBGain calculated by WpType1

⑧ wbGainType3 is WBGain calculated by WpType3, wbWeightType3 is the weight (WpType1 is 1-wbWeightType3)

⑨ stategy_result.gain is WBGain under each light source

⑩ prob_total(0.372926),prob_dis(0.151197),prob_LV(0.142857),prob_WPNO(0.383987) respectively represent the total probability of each light source, the probability of distance, the probability of brightness, the probability of the number of white points

⑪ spatial gain(rggb):(1.745900,1.000000,1.000000,1.824126), statistics gain weigth(1.000000) respectively represents the dayGain of each light source (strategy WBGain), and each light source is based on the statistical white point output WBGain probability, then Strategy WBGain probability is 1-statistics gain weigth

⑫ type0: gain (rg,bg):(1.287584,2.843158) WPNo(7707) is the white balance gain calculated by the middle frame, the number of white points

⑬ type1: gain (rg,bg):(1.291619,2.862351) WPNo(8557) is the white balance gain calculated by the large frame, the number of white points

⑭ type2: gain (rg,bg):(1.288622,2.825095) WPNo(6905) Weight(0.895939) is the white point gain, the number of white points, and the weight involved in the calculation of WBGain

⑮ excWpRangeResult[0]: gain (rg,bg):(0.000000,0.000000) WPNo(0),mode(0) respectively represent the white balance gain of the 0th non-white point interval or the white point interval of the additional light source, white Number of points

(3) Other

①

```
: Global
: CCT:4342.158203,CCRI:0.072367

: ill:A  prob: 0.372926
: CCT:3058.074463,CCRI:0.034103

: ill:D50  prob: 0.369272
: CCT:5710.621582,CCRI:0.096392

: ill:TL84  prob: 0.233994
: CCT:4246.391602,CCRI:0.107421

: ill:CWF  prob: 0.014538
: CCT:4403.761719,CCRI:-0.060823
```

This information is used to characterize the color temperature of the scene. It is only auxiliary information. Global is the comprehensive color temperature. Ill lists the probability and color temperature of the first few light sources that participate in the calculation of WBGain.

Locate the problem from the log

① If the WPNo value is relatively small and there are some white spots in the actual scene, the white spot conditions need to be re-adjusted

② Look at whether the statistics gain weigth of each light source is 1, and whether it is affected by the strategy gain

③ Check whether the weight of wbWeightType3 is 1:

If the scene does have fewer white points, but wbGainType3 may be more consistent with the actual color temperature, you need to adjust WP_THL, WP_THH to divide the current scene into WpType3

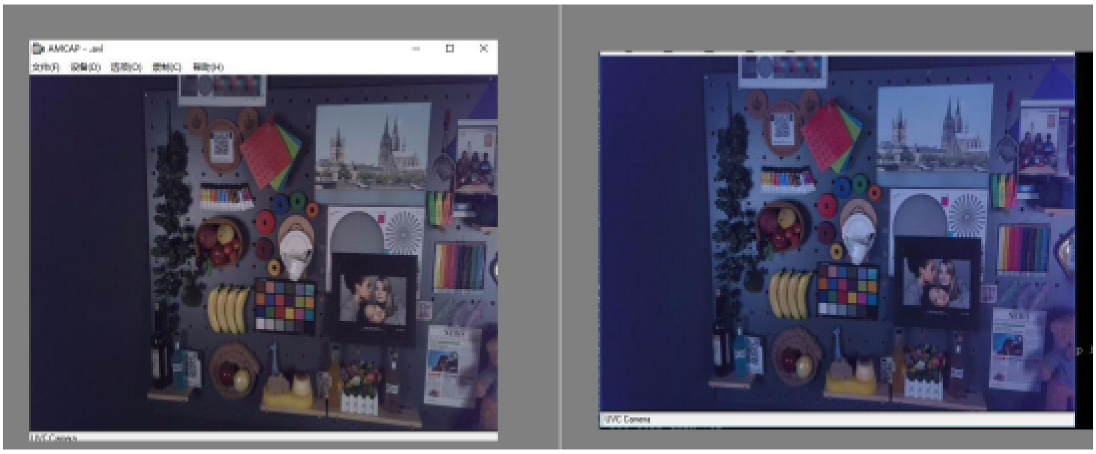
In other cases, please refer to the previous two points for adjustment

Catch raw and simulate

When you need to re-adjust the white point conditions, or when you can't locate the problem from the Log, you need to grab the raw image to simulate the white point detection and view the white point statistics under each light source.

(1) Example 1

On the right is the problematic scene, and on the left is the effect of grabbing the raw image and re-adjusting the white point conditions.



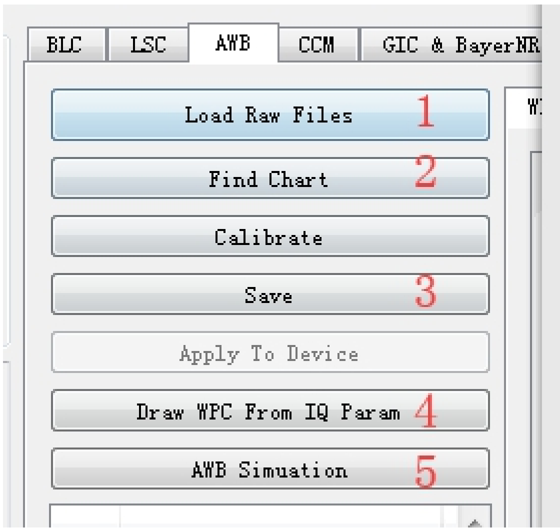
Steps:

① Grab raw picture 1

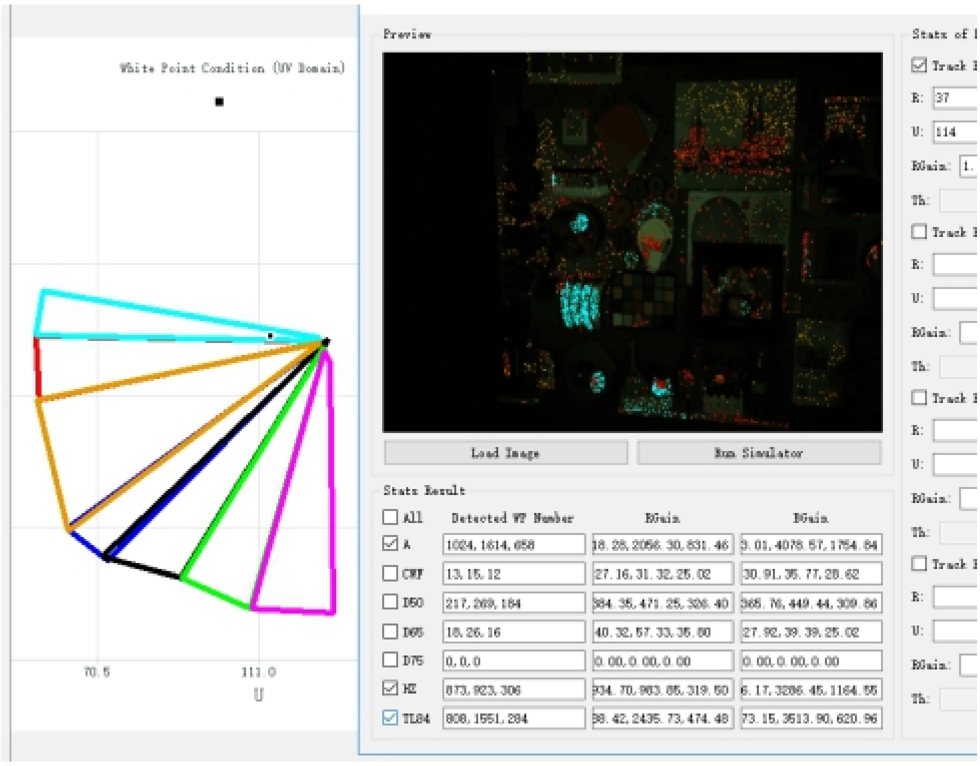
② Open RKISPCalibrationTool and import the xml file

③ BLC LSC parameters are configured, the current version needs to re-calibrate these two parameters

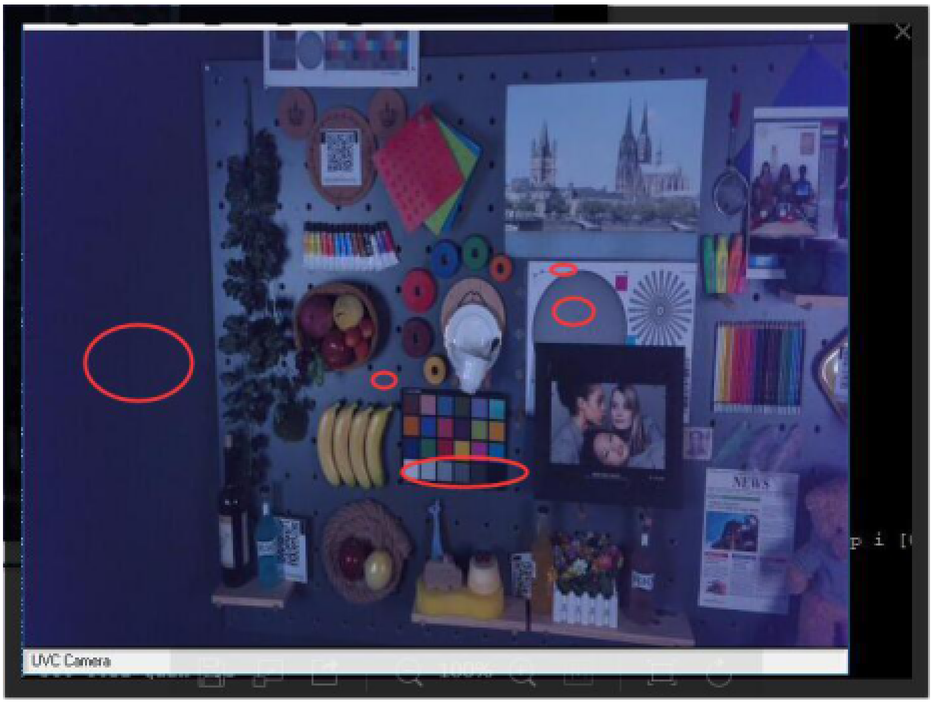
④ In order to refer to the distribution of white and non-white patches of the color card under the previous standard light source, you need to re-Load Raw Files to import all the graphs used during calibration, click FindChart to complete the color card identification, do not click Calibrate, click Save, click Draw WPC From IQ Param to import the white point conditions, and display the color patch distribution of the color card.



⑤ Click Load Image on the AWB Simulation interface to import raw picture 1, the picture and white point detection results will be updated on the interface at the same time, as shown below



It can be seen that there are very few white spots identified, and the picture is indeed darker. As shown below, the gray and white areas similar to the circle should be detected as white dots



⑥　Click the gray area on the image,

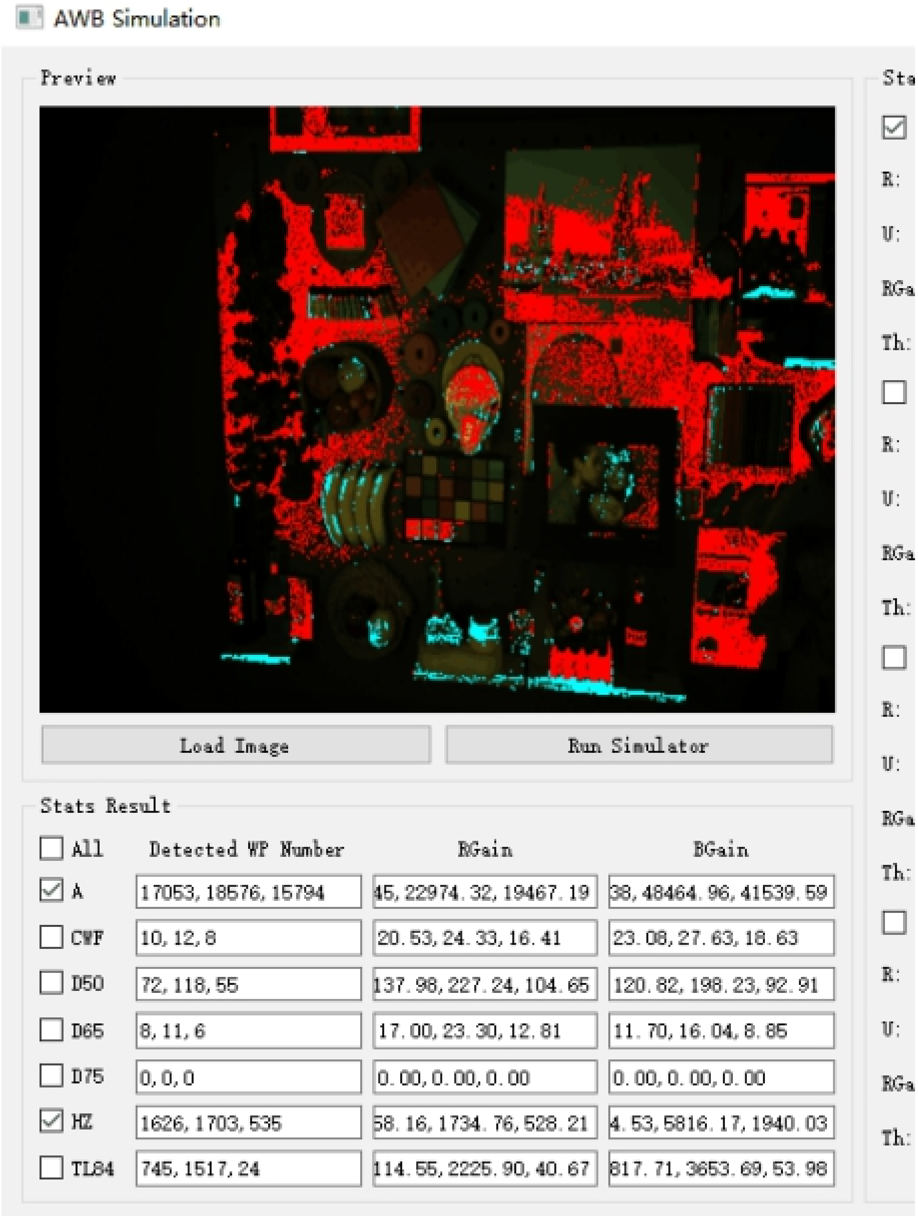
Check whether the mapped point is within the white point interval of XY and UV. If not, adjust the white point condition to make it fall within the interval;

The AWB Simulaton interface will also display the value of R G B and Y at this point. It needs to be compared with the limitRange set in the XML to see if it exceeds the range. If this is the case, you can appropriately relax the point limitRange range. Because the particularly dark and bright points are affected by noise or saturation of a certain channel, their Rgain Bgain will deviate from the actual one, so this range needs to be weighed;

If it is within the range of limitRange and within the white point range of XY and UV, but it is not recognized as a white point, you need to confirm the TH of YUV. The th of this point will also be displayed on the AWB Simulaton interface. Refer to the interface Adjust the TH of YUV with this value on the above;

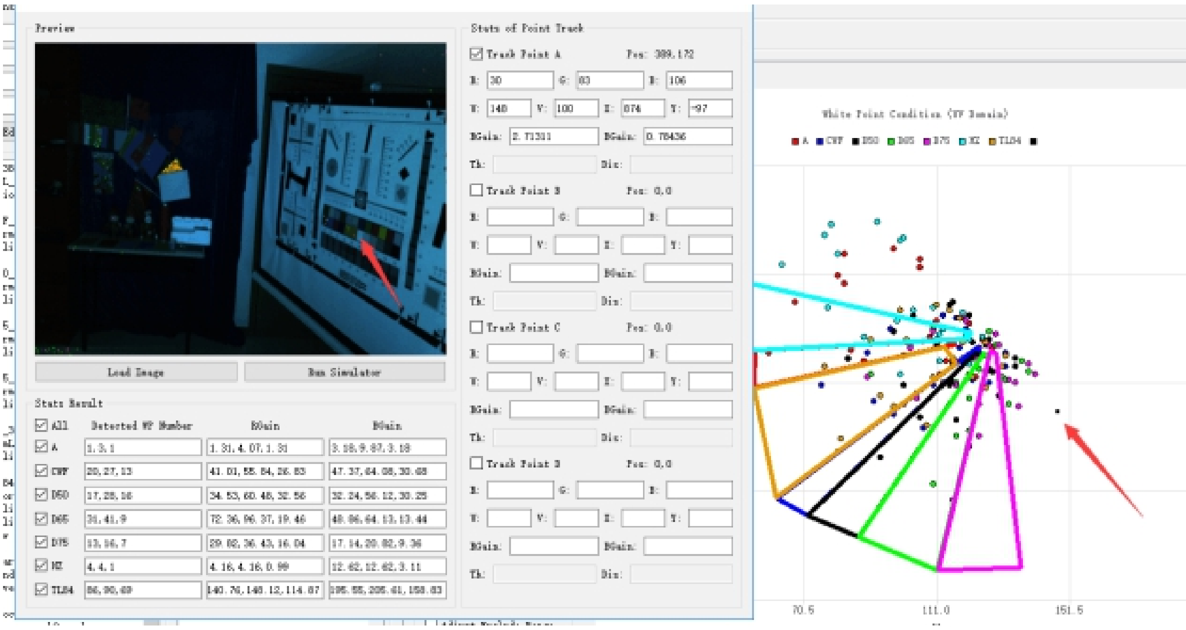
⑦　After adjusting, click Save on the AWB interface, and click Run Simlation on the AWB Simulation interface to re-simulate.

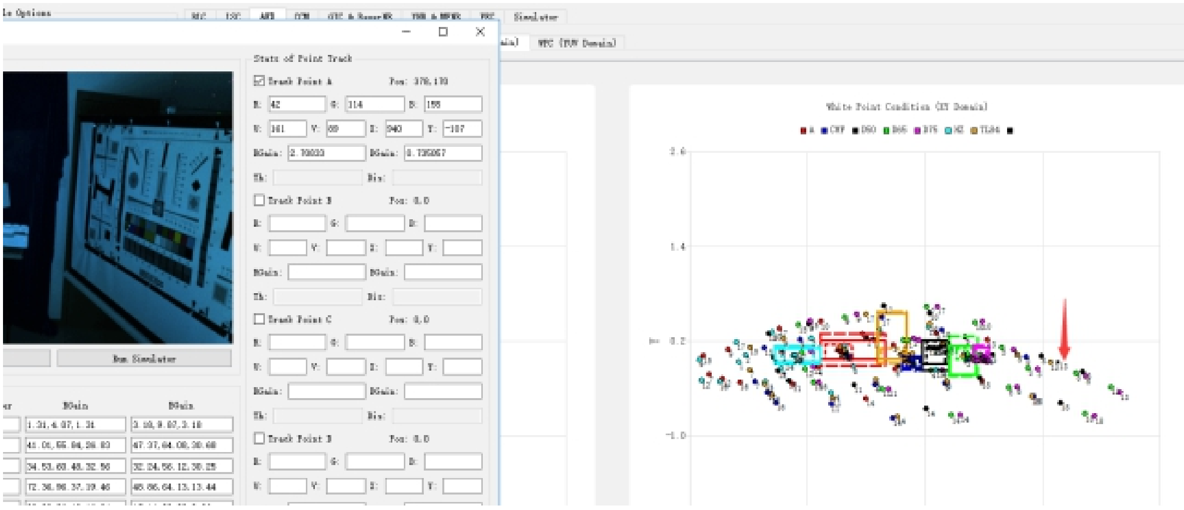
After modifying the white point interval of XY and UV, the white point detection is as follows, which solves the problem of abnormal white balance



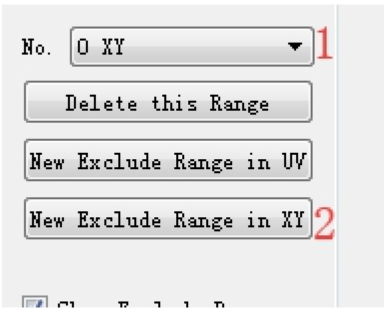
(2) Example 2

For the TV camera, the special feature is that the light source of the scene is greatly affected by the light source of the TV screen and the content of the TV screen. The following shows the phenomenon of abnormal white balance of the TV camera.

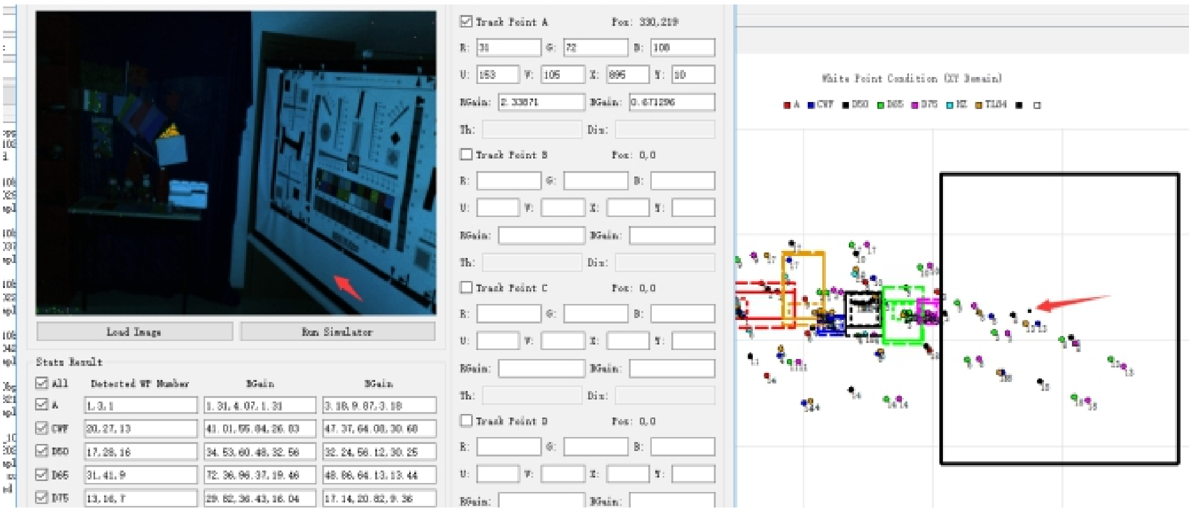




The red arrow points out the mapping position of the gray point in the XY domain and the UV domain. It can be seen that the deviation from the normal white point is farther at this time. Solid can add extra white point interval to include these light sources,



As follows



It should be noted that you need to manually modify the mode to 3

```
<excludeRange index="1" type="cell" size="[1 7]">
  <cell index="1" type="struct" size="[1 1]">
    <Domain index="1" type="double" size="[1 1]">
      [2 ]
    </Domain>
    <mode index="1" type="double" size="[1 1]">
      [1 ]
    </mode>
    <window index="1" type="double" size="[1 4]">
      [457 1222 160 -286 ]
    </window>
  </cell>
  <cell index="2" type="struct" size="[1 1]">
```

When the default mode = 1, treat this interval as a non-white point interval

(3) To be added

Examples of special problems

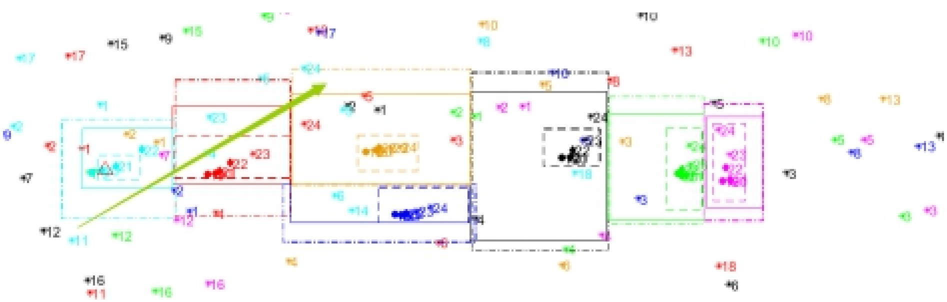
(1) In the webcam application, the white balance is set to natural light mode, which causes the automatic white balance to not be turned on.

By looking at the log, it is found that the current mode=1 is manual mode

```
[09:51:39.57935][AWB]:XCAM DEBUG rk_aiq_awb_algo_v200.cpp:2795: AwbReConfigV200: byPass: 0  mode:1
```

This problem can be solved by changing the white balance mode

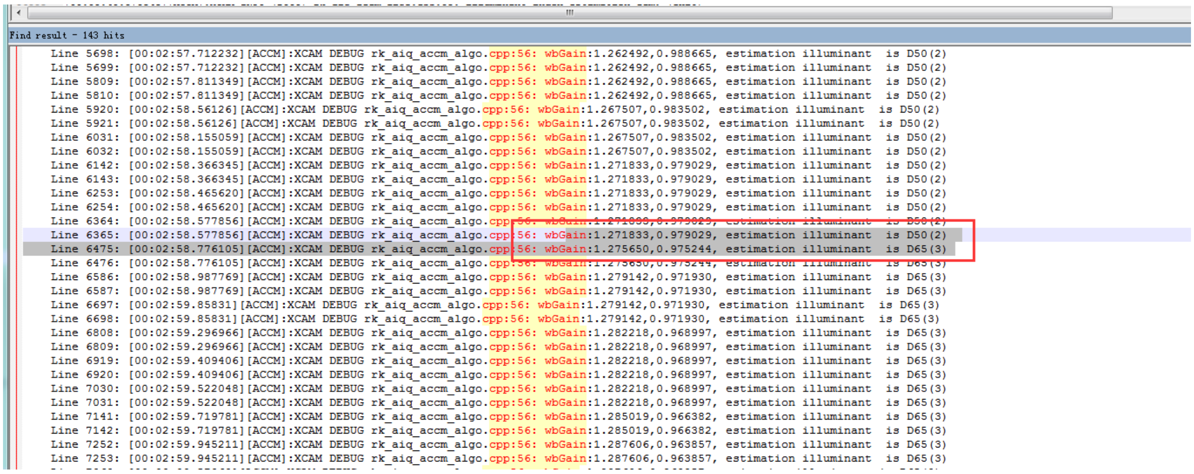
(2) When calibrating, it is found that the distribution of white points under hz a light is not concentrated



The actual effect is as follows, the white balance is wrong

Later tests found that the near-infrared band was not cut off because the infrared filter was unqualified, which was solved by replacing the infrared filter.

(3) The light source of the cc module is estimated to be oscillating back and forth, causing the color to oscillate



```
Find result - 143 hits
Line 5698: [00:02:57.712232][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.262492,0.988665, estimation illuminant  is D50(2)
Line 5699: [00:02:57.712232][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.262492,0.988665, estimation illuminant  is D50(2)
Line 5809: [00:02:57.811349][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.262492,0.988665, estimation illuminant  is D50(2)
Line 5810: [00:02:57.811349][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.262492,0.988665, estimation illuminant  is D50(2)
Line 5920: [00:02:58.56126][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.267507,0.983502, estimation illuminant  is D50(2)
Line 5921: [00:02:58.56126][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.267507,0.983502, estimation illuminant  is D50(2)
Line 6031: [00:02:58.155059][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.267507,0.983502, estimation illuminant  is D50(2)
Line 6032: [00:02:58.155059][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.267507,0.983502, estimation illuminant  is D50(2)
Line 6142: [00:02:58.366345][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6143: [00:02:58.366345][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6253: [00:02:58.465620][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6254: [00:02:58.465620][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6364: [00:02:58.577856][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6365: [00:02:58.577856][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.271833,0.979029, estimation illuminant  is D50(2)
Line 6475: [00:02:58.776105][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.275650,0.975244, estimation illuminant  is D65(3)
Line 6476: [00:02:58.776105][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.275650,0.975244, estimation illuminant  is D65(3)
Line 6586: [00:02:58.987769][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.279142,0.971930, estimation illuminant  is D65(3)
Line 6587: [00:02:58.987769][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.279142,0.971930, estimation illuminant  is D65(3)
Line 6697: [00:02:59.85831][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.279142,0.971930, estimation illuminant  is D65(3)
Line 6698: [00:02:59.85831][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.279142,0.971930, estimation illuminant  is D65(3)
Line 6808: [00:02:59.296966][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 6809: [00:02:59.296966][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 6919: [00:02:59.409406][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 6920: [00:02:59.409406][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 7030: [00:02:59.522048][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 7031: [00:02:59.522048][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.282218,0.968997, estimation illuminant  is D65(3)
Line 7141: [00:02:59.719781][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.285019,0.966382, estimation illuminant  is D65(3)
Line 7142: [00:02:59.719781][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.285019,0.966382, estimation illuminant  is D65(3)
Line 7252: [00:02:59.945211][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.287606,0.963857, estimation illuminant  is D65(3)
Line 7253: [00:02:59.945211][ACCM]:XCAM DEBUG rk_aiq_accm_algo.cpp:56: wbGain:1.287606,0.963857, estimation illuminant  is D65(3)
```

By increasing the value value in the tolerance node in xml, when the change of wbgain is smaller than this value, it will not be updated to achieve the goal of stability. In this example, modify the parameters in xml to

```
<tolerance index="1" type="struct" size="[1 1]">
    <LV index="1" type="double" size="[1 4]">
        [0  64 128 256.0000 ]
    </LV>
    <value index="1" type="double" size="[1 4]">
        [0.05 0.05 0.05 0.05 ]
    </value>
</tolerance>
```

3 Basic color adjustment CC

3.1 Function description

Since the sensor spectrum distribution function is difficult to completely match the visual response function, a color correction matrix (Color Correction Matrix, CCM) can be used to correct the cross effect and response intensity of the spectral response, so that the image captured by the front end can maintain the color of the human vision Unanimous.

The CCM calibration tool supports 3x3 CCM (aij) pre-calibration for 24 color cards.

$$\begin{bmatrix} R_{cc} \\ G_{cc} \\ B_{cc} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix}$$

RV1109 supports multiple groups of CCMs with different color temperatures. When ISP2.0 is running, the gain node can be configured according to the IQ parameters to adjust the global saturation or local saturation to realize the dynamic adjustment of the CCM matrix coefficients.

$$\begin{bmatrix} R_{cc} \\ G_{cc} \\ B_{cc} \end{bmatrix} = alpha * scale * \begin{bmatrix} a_{11}-1 & a_{12} & a_{13} \\ a_{21} & a_{22}-1 & a_{23} \\ a_{31} & a_{32} & a_{33}-1 \end{bmatrix} \cdot \begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix} + \begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix}$$

3.2 Key parameters

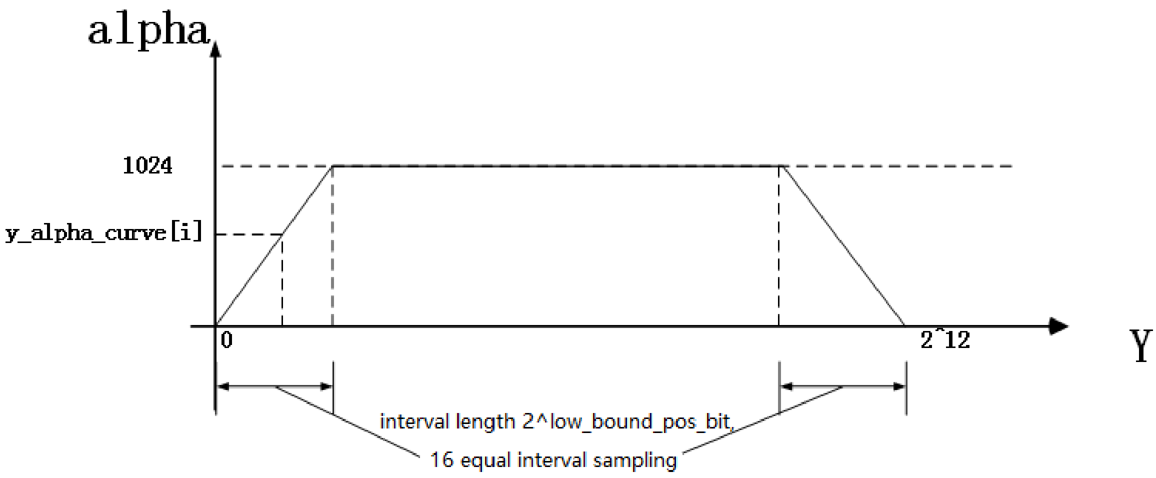
Enable control

Parameters	Description
enable	Color correction enable switch, 1 means enable; the value is 0 or 1
damp_enable	Color correction matrix smoothing function switch, 1 means use this function; value 0 or 1

Brightness-saturation adjustment

Pixel brightness related saturation adjustment

Parameters	Description
RGB2Y_para	Calculation coefficient from RGB to Y, 7bit fixed-point value; integer, value range [0,128]
low_bound_pos_bit	Pixel brightness (Y)-color correction (CC) intensity brightness threshold; integer, value range [0,10]
y_alpha_curve	Pixel brightness (Y)-color correction (CC) intensity (alpha) intensity, 1024 means 1 times intensity, 0 means no correction; integer, value range [0,1024]



Y-CcAlpha The length of the left and right intervals is equal to 2 to the (low_bound_pos_bit)th power,both of them are divided equally into 16 segments

Global saturation adjustment

Different exposure gains (gain) correspond to different correction intensity scales, corresponding to the parameters under the node gain_alphaScale_curve

Parameters	Description
gain	gain-scale exposure gain component, decimal, value greater than 0
scale	The corrected intensity component of gain-scale, decimal, value range [0,1]

CCM select control parameters

The parameters of the corresponding light source are automatically selected according to the white balance gain, and different exposure gains (gain) corresponding to different saturation (sat) CCMs can be configured under a certain light source

Parameters	Description
name	Light source name
wbGain	The standard white balance gain corresponding to the light source, decimal, generated by the calibration tool, and the value is greater than 0
matrixUsed	CCM that will be used under this light source
gains	Exposure gain component of gains-sat, decimal, value greater than 0
sat	The saturation component of gains-sat, as a decimal, the value is greater than 0

CCM parameters

Parameters	Description
Name	CCM Name
illumination	light source name
saturation	The corresponding saturation, which is generated by the calibration tool, and the value is greater than or equal to 0
ccMatrix	Color correction matrix, generated by the calibration tool, decimal, value range [-8, 7.992]
ccOffsets	R\G\B component offset, generated by calibration tool, value range [-4095-4095], generally not used

3.3 CCM calibration

Complete the CCM calibration work according to "Rockchip IQ Tools Guide ISP2x v1.1".

RAW data collection

Calibration light source selection

Seven light sources with different color temperatures: D50, D65, D75, A, CWF, HZ, TL84

Collection steps

Step 1. Place the color card in the center of the background wall of the light box to ensure that the light sources on the left and right sides are uniform; if the project has high color requirements, you can also put the corresponding color next to it, such as a skin color card, to confirm the effect.

Step 2. Adjust the exposure so that every color block of the color card after applying gamma cannot be overexposed. It is recommended to use automatic exposure

Step 3. When shooting, adjust the object distance so that the proportion of the color card on the screen is 1/9.

Calibration

Step

Step 1. For RAW data import and selection of 24-color area, please refer to "Rockchip IQ Tools Guide ISP2x v1.1", Part 4, Module 5 "CCM".

Step 2. Configure calibration parameters

(1) Set gamma: select the gamma curve that the camera will use. Supports Normal, HDR, Night modes, and also supports customization.

(2) Set the color block weight: configure the color block weight in the 6x4 table, and the color block position corresponds to the position in the table.

(3) Click the "Calibrate" button to calibrate and obtain CCM. Saturation can be manually adjusted on the Calibrate page until the corrected effect diagram or color difference diagram in the Result meets the requirements.

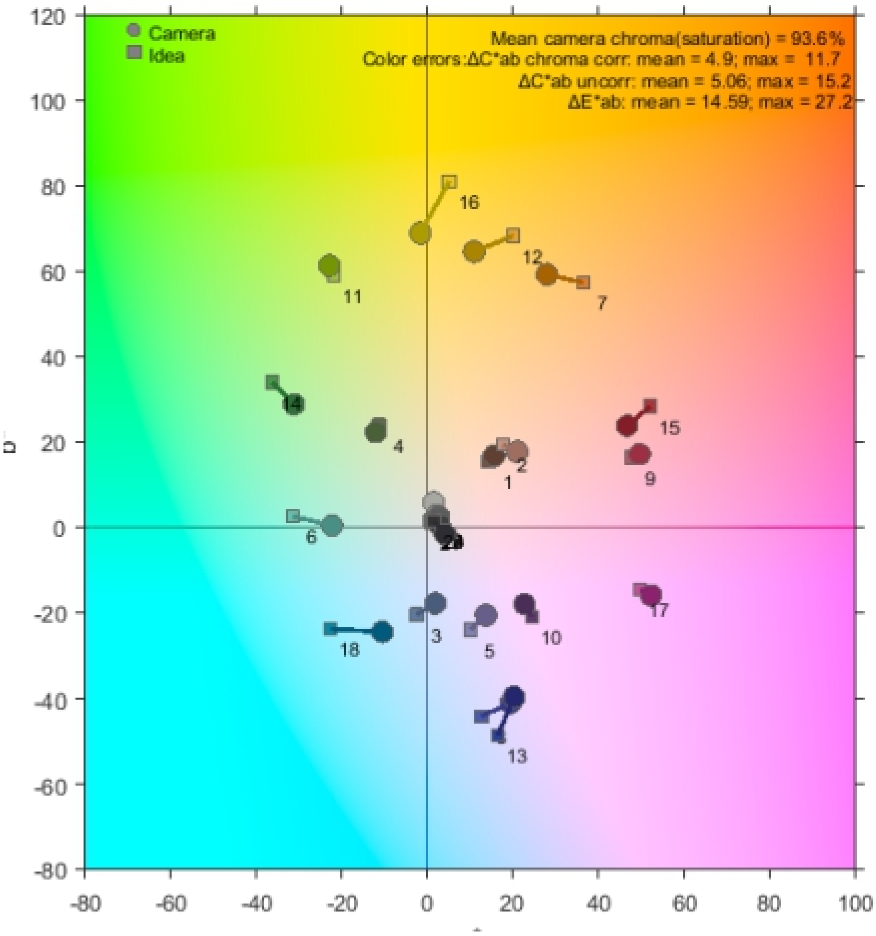
Chromatic aberration chart introduction

According to the deviation direction and the interval of the standard color block in the color difference chart, analyze which component is abnormal, as follows:

(1) The color block of the camera is farther from the origin than the color block of the idea, and the saturation of the camera is higher than that of the idea

(2) The distance between the color block of the camera and the origin is smaller than the color block of the idea, and the saturation of the camera is higher than that of the idea.

Case picture:



The camera saturation of color block 15 (red block) and color block 14 (green) is lower than idea, but it belongs to the category of relatively small deviation

Color block 13 (blue) is biased toward purple, and human vision may also feel that color block 13 is also biased toward purple, so this deviation is acceptable.

Generally, it is necessary to ensure that the deviation of the 13-15 color block color block is not too large, which almost represents the three primary colors, and other colors can be obtained from the superposition of these three colors.

If the 13-15 color block, or other color blocks that are more concerned, have a serious color cast, you can increase the weight of the color block, but pay attention to the impact on the color cast of other color blocks.

Precautions

(1) When identifying the 24-color area, make sure that the black border of each color block is not selected

(2) It may be necessary to re-calibrate the CCM after adjusting the gamma curve, so it is better to adjust the gamma first

(3) Improper brightness of the calibration chart will affect the saturation characteristics of the calibrated CCM. The CCM saturation specified by the over-bright RAW icon is low, and the CCM saturation specified by the too dark RAW icon is high.

(4) The recommended objective indicators are as follows, but they can vary from project to project, and do not pay attention to these objective indicators

Color accuracy	D65(external)	color saturation	110-120%
		mean(ΔC)	<10
		max(ΔC)	<20
		mean(ΔE)	<15
	TI84 (for internal only)	color saturation	110-120%
		mean(ΔC)	<10
		max(ΔC)	<20
		mean(ΔE)	<12
	Coolwhite (for internal only)	color saturation	110-125%
		mean(ΔC)	<10
		max(ΔC)	<20
		mean(ΔE)	<12
	A light (for internal only)	color saturation	110-120%
		mean(ΔC)	<10
		max(ΔC)	<22
		mean(ΔE)	<12

3.4 Color adjustment

Overall color saturation adjustment

Adjust the parameters of gain_alphaScale_curve

The value of scale can be adjusted within the range of [0, 1.0] to affect the final color correction intensity. The smaller the scale, the lower the color saturation, and vice versa.

Adjust gains-sat

The smaller the sat, the lower the color saturation, and vice versa, the color saturation increases. Different light sources can adjust different parameters.

Increase CCM with high saturation

When the first two points are adjusted to the maximum value, the saturation is still not enough. It is necessary to recalibrate the CCM with higher saturation, and at the same time, adjust the maximum value of sat in gains-sat to the increased saturation.

Reduce the color saturation of dark pixels

Decrease the value in y_alpha_curve to reduce the color saturation of dark pixels. Note that the last value must be 1024, otherwise it will affect the color saturation of other brightness in the image.

Some color adjustments

After the overall color saturation adjustment is completed, the color still does not achieve the expected effect. You can try the following steps:

(1) When it is necessary to adjust the color to be consistent with human vision, confirm whether the white balance is correct;

(2) When it is necessary to adjust the color to be consistent with the contrast machine, confirm whether the white balance is consistent;

(3) When it is necessary to adjust the color to be consistent with the contrast machine, confirm whether the brightness is close to

(4) If the white balance is confirmed to be consistent or correct and the brightness is close, the color still does not meet the expectations, then adjust the CCM related parameters to achieve the goal.

Confirm whether the white balance is correct

Key points: Whether the white object is color cast.

Method:

Look with your eyes, whether the white objects in the video are white;

Snap a picture to see if the R/G/B components of the white block are quite different.

Confirm whether the white balance is consistent with the contrast machine and adjust

(1) If the white balance of the contrast machine is relatively correct, and the white object of RK has a more obvious color cast, first use the white balance module to make the white balance more correct;

(2) If the white object of the contrast machine has a more obvious color cast and the white balance of RK is relatively correct, it is necessary to distinguish whether it is caused by the defect of the white balance algorithm of the contrast machine or the color preference of the contrast machine; if the color preference is different, you can First adjust the hue through the white balance module to make the two consistent, or use tools such as faststone to adjust the hue of the contrast machine to be the same as that of RK; if it is caused by a defect in the white balance algorithm of the contrast machine, you can increase the number of white points in the scene and re-capture the picture. Or use faststone and other tools to adjust the tone of the contrast machine to be the same as that of RK.

(3) How to distinguish the white objects of the contrast machine, the obvious color cast is caused by the defect of the white balance algorithm of the contrast machine, or the color preference of the contrast machine is different

a. If the scene has only white objects and the brightness is appropriate, the white of the contrast machine is still a color cast, which is most likely because the contrast machine has adjusted the hue;

b. Otherwise, it is caused by a defect in the algorithm of the comparison machine;

Adjust the brightness to be consistent with the contrast machine

(1) Adjust the brightness related modules (ae, gamma, dehaze, hdr) to make the brightness close, allowing a certain gap

(2) Or use faststone and other tools to adjust the brightness of the contrast machine to be the same as that of RK

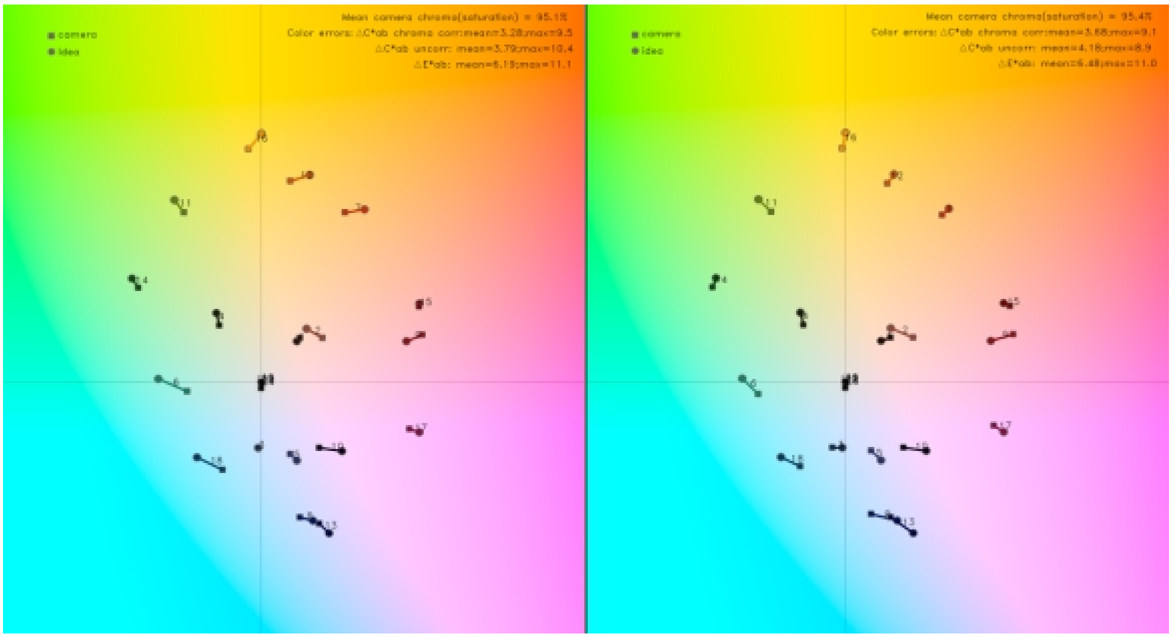
Adjust CCM

Re-calibrate CCM with tools

Find the color block closest to the color in the 24 color card, increase the weight of the color block, and re-calibrate the CCM.

Case:

In the color difference diagram on the left, the color shift of the 6th/18th color block is relatively large. For this, the weight of the neutral color block (19th to 24th color block) can be set to 0, the 6th color block is set to 16, and the weight of the neutral color block (19th to 24th color block) is set to 0. The weight of the 18 color block is set to 8. In addition, in order to reduce the impact of the above adjustments on other color blocks, the weight of the three primary color blocks (the 13th-15th color blocks) is set to 8, and the weight of the skin color block (the second color block) is also set to 8, so the result is obtained The color difference diagram is as shown on the right, and the color cast of the 6th/18th color block is reduced.



Manually adjust CCM

Get RK RGB value

Use the RK machine to capture the image and get the RK RGB value

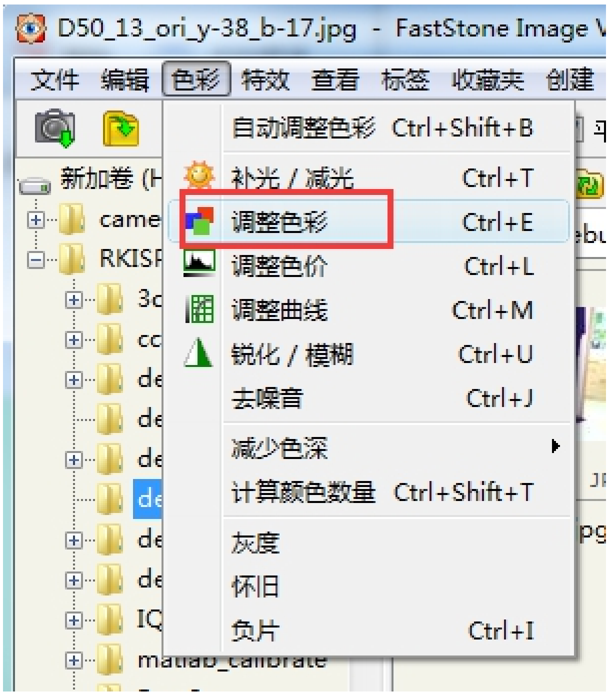
Get the target RGB value

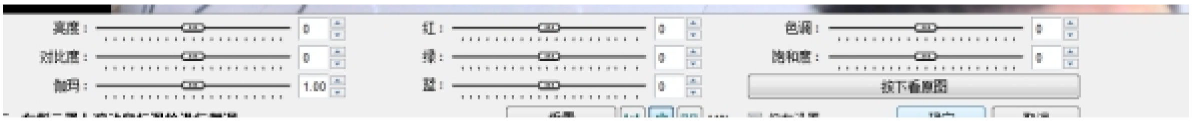
(1) When there is a comparison machine

Use the contrast machine to capture the image to obtain the target value, but the brightness and white balance of the RK need to be close

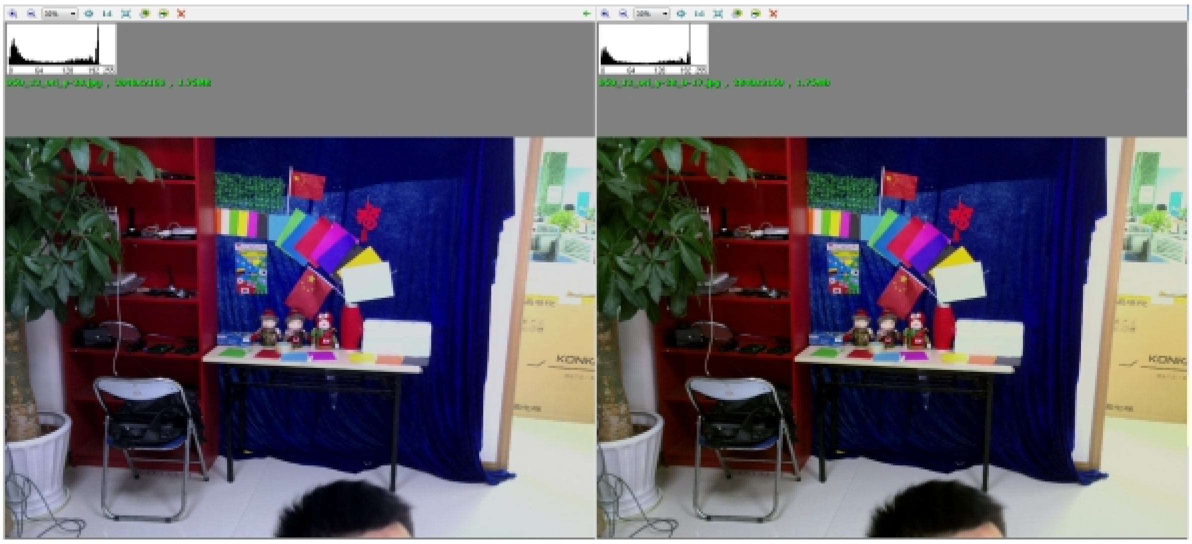
(2) When there is no contrast machine

Use tools such as faststone to adjust a certain color of interest on the RK acquisition map until the color matches the expected color.





For example: adjust the B component as the target value, use fatone to subtract 17 from the B channel, the green shown in the lower right is the expected color,



Compare at this time the green RGB of RK is 64 85 90, and the target is 65 86 69. Then you know that you need to adjust the CCM to reduce the B component, then the two greens will be close.

(3) The target value of 24 color card human vision is:

No.	Number	sRGB			CIE L*a*b*			Munsell Notation Hue Value / Chroma	
		R	G	B	L*	a*	b*		
1.	dark skin	115	82	68	37.986	13.555	14.059	3 YR	3.7 / 3.2
2.	light skin	194	150	130	65.711	18.13	17.81	2.2 YR	6.47 / 4.1
3.	blue sky	98	122	157	49.927	-4.88	-21.925	4.3 PB	4.95 / 5.5
4.	foliage	87	108	67	43.139	-13.095	21.905	6.7 GY	4.2 / 4.1
5.	blue flower	133	128	177	55.112	8.844	-25.399	9.7 PB	5.47 / 6.7
6.	bluish green	103	189	170	70.719	-33.397	-0.199	2.5 BG	7 / 6
7.	orange	214	126	44	62.661	36.067	57.096	5 YR	6 / 11
8.	purplish blue	80	91	166	40.02	10.41	-45.964	7.5 PB	4 / 10.7
9.	moderate red	193	90	99	51.124	48.239	16.248	2.5 R	5 / 10
10.	purple	94	60	108	30.325	22.976	-21.587	5 P	3 / 7
11.	yellow green	157	188	64	72.532	-23.709	57.255	5 GY	7.1 / 9.1
12.	orange yellow	224	163	46	71.941	19.363	67.857	10 YR	7 / 10.5
13.	blue	56	61	150	28.778	14.179	-50.297	7.5 PB	2.9 / 12.7
14.	green	70	148	73	55.261	-38.342	31.37	0.25 G	5.4 / 8.65
15.	red	175	54	60	42.101	53.378	28.19	5 R	4 / 12
16.	yellow	231	199	31	81.733	4.039	79.819	5 Y	8 / 11.1
17.	magenta	187	86	149	51.935	49.986	-14.574	2.5 RP	5 / 12
18.	cyan	8	133	161	51.038	-28.631	-28.638	5 B	5 / 8
19.	white (.05*)	243	243	242	96.539	-0.425	1.186	N	9.5 /
20.	neutral 8 (.23*)	200	200	200	81.257	-0.638	-0.335	N	8 /
21.	neutral 6.5 (.44*)	160	160	160	66.766	-0.734	-0.504	N	6.5 /
22.	neutral 5 (.70*)	122	122	121	50.867	-0.153	-0.27	N	5 /
23.	neutral 3.5 (1.05*)	85	85	85	35.656	-0.421	-1.231	N	3.5 /
24.	black (1.50*)	52	52	52	20.461	-0.079	-0.973	N	2 /

Adjust CCM description

Compare the current RGB and target RGB values, and manually adjust the CCM to make the two RGB close.

(1) CCM adjustment constraints

The formula of the color correction matrix is as follows:

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

In order to ensure that the white balance is not damaged, the parameters must meet the conditions:

$$a_{i1} + a_{i2} + a_{i3} = 1$$

Each channel mainly comes from the color component of the original channel, so the conditions must be met:

$$a_{ii} \geq 1$$

At the same time, try to make the difference of the element value of the main diagonal line as small as possible, and try to make the elements outside the main diagonal line have negative values.

If a_{13} is a positive number, it will cause high saturation red to become purple. If a_{31} is a positive number, it will cause high saturation blue to become purple.

When a_{21} is a negative value, the greater the absolute value, the smaller the G component value of the red after correction, and the higher the red saturation; a_{23} is a negative value When the absolute value is larger, the G component value of the corrected blue is smaller, and the saturation of the blue is higher.

(2) Summary of fine adjustment of common color casts:

Blue (red) is slightly purple. When a_{13} (a_{31}) is positive, the R (B) component needs to be reduced, and a_{13} (a_{31}) changed from a positive number close to 0 to a smaller negative number;

Blue (red) is oversaturated, when a_{23} (a_{21}) is negative, the G component needs to be increased, and a_{23} can be reduced a_{21} absolute value;

Purple is bluish, need to increase R component, increase a_{13}, decrease a_{11} and a_{12}; or decrease B component , Decrease a_{33}, increase a_{31} and a_{32};

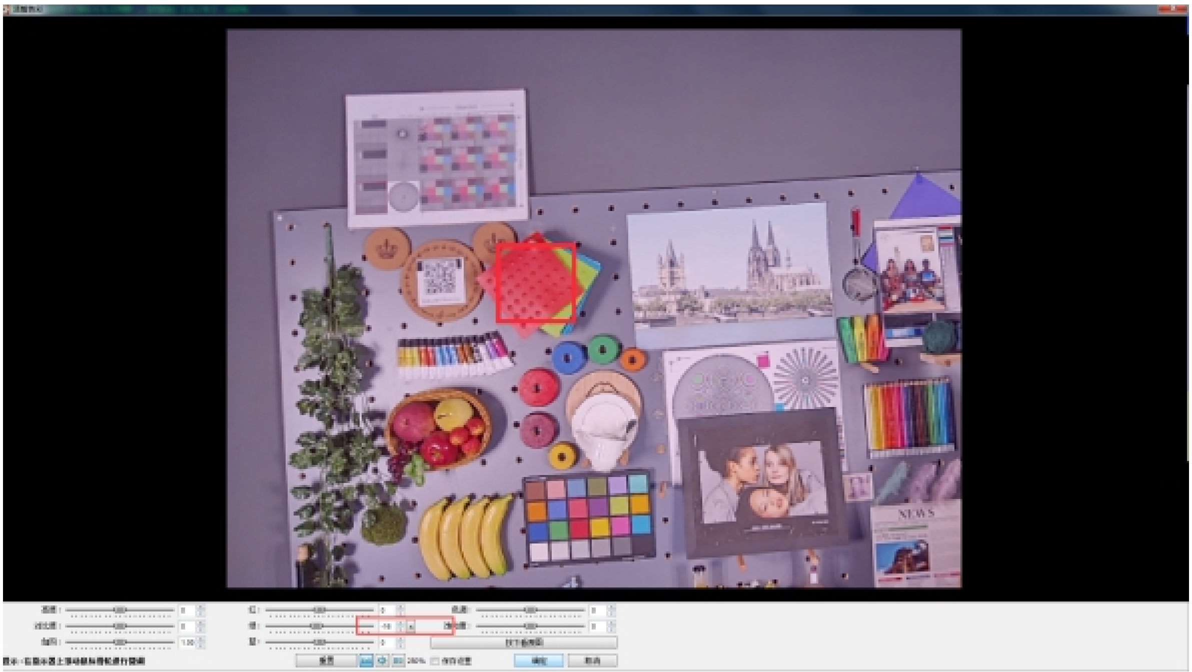
The red is orange, and the G component needs to be reduced, which can reduce a_{21} and increase a_{22};

The skin color is yellowish green. It is necessary to reduce the G component and increase the B component, which can greatly reduce a_{22} and increase a_{23}, and fine-tune a_{21}, greatly increase a_{31} and decrease a_{32}, fine-tune a_{33}.

Adjust CCM example

(1) Case 1 Red to orange:

Use faststone to adjust RGB and found that reducing the G component can improve the orange problem. At this time, the target RGB value of the red plastic film is [212 63 79].



The red plastic sheet in the red frame is orange, RGB = [212, 78, 80], compared with the target value [212 63 79], the G component is too large. If you have more experience, you can skip the step of obtaining faststone and directly adjust the CCM to reduce the G component.

$$G' = a_{21}R + a_{22}G + a_{23}B ,$$

Original correction factor: $[a_{21}, a_{22}, a_{23}] = [-0.2854, 1.1496, 0.1358]$

Since the R component value of the red plastic sheet is the largest, the value of a_{21} needs to be reduced. In order to meet the constraint that the rows add to 1, the absolute value of a_{22} needs to be reduced

Correction coefficient after adjustment: $[a_{21}, a_{22}, a_{23}] = [-0.385, 1.2497, 0.1358]$

Red plastic sheet: RGB = [208, 56, 76].

(2) Case 2 The skin tone is yellowish green:

The skin color in the red frame is yellowish green, RGB = [216, 174, 124], where the G component is too large and the B component is too small;

Original correction factor: $\begin{bmatrix} a_{21}, a_{22}, a_{23} \\ a_{31}, a_{32}, a_{33} \end{bmatrix} = \begin{bmatrix} -0.3192 & 1.6927 & -0.3735 \\ 0.0239 & -0.5738 & 1.5499 \end{bmatrix}$ Same as case 1, reduce the G component, Significantly reduce the absolute value of a_{22} and a_{23} de, and fine-tune a_{21},

Skin tone at the corresponding position ~~at this time~~: RGB = [212, 169, 124];

~~In order to increase the B component, because~~ the R and G components have large values, ~~so~~ significantly reduce the absolute value of a_{31} and a_{32}, and fine-tune a_{33}

Correction coefficient after adjustment: $\begin{bmatrix} a_{21}, a_{22}, a_{23} \\ a_{31}, a_{32}, a_{33} \end{bmatrix} = \begin{bmatrix} -0.3004 & 1.6375 & -0.3371 \\ 0.2127 & -0.7294 & 1.5166 \end{bmatrix}$;

The skin color at the corresponding position ~~at this time~~: RGB = [214, 169, 146].

4 Advanced color adjustment-3DLut

4.1 CCM VS 3DLut

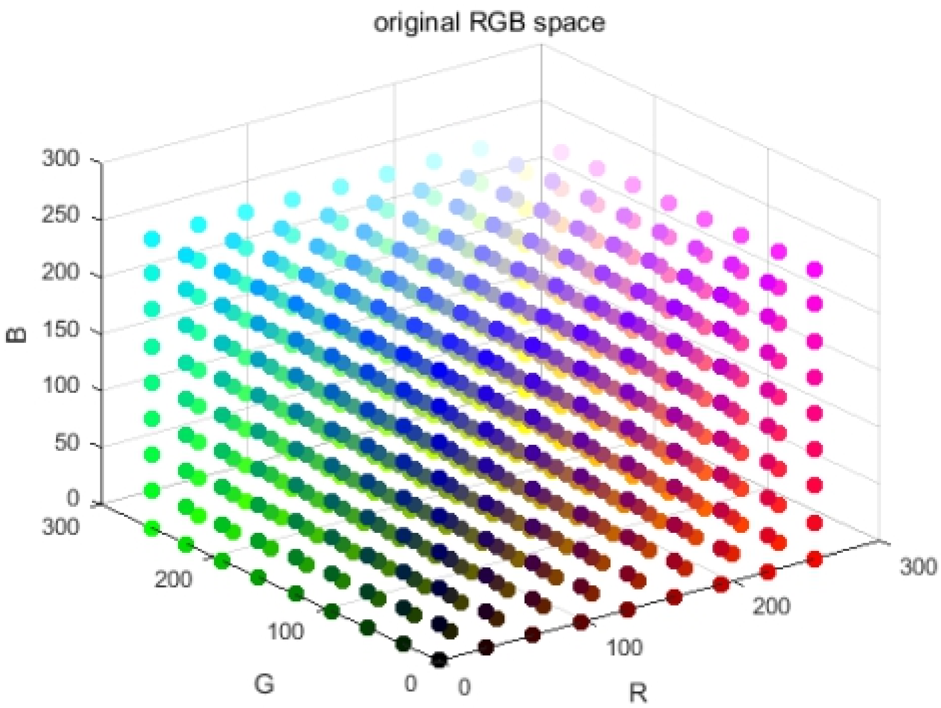
The task of CCM is to make the colors under different light sources similar to human vision, and the task of 3DLut is to adjust individual colors according to preferences. The advantages and disadvantages of the two for color adjustment are as follows:

	CCM	3DLut
Advantages	The color transition is natural, and it is not easy to cause noise	It is easier to adjust the hue and saturation of a color; it has no effect on the colors that are not similar
Disadvantages	Modifying the CCM for a certain preference may cause other dissimilar colors to be affected; color adjustment is more difficult	Because the current number of sampling points 9x9x9 is small, the color is easy to transition and unnatural, and the adjustment of the value will affect the pixels Denoising intensity of a point, which will introduce noise

As for which solution to choose, the actual project is directly weighed against the color preference and transition and noise.

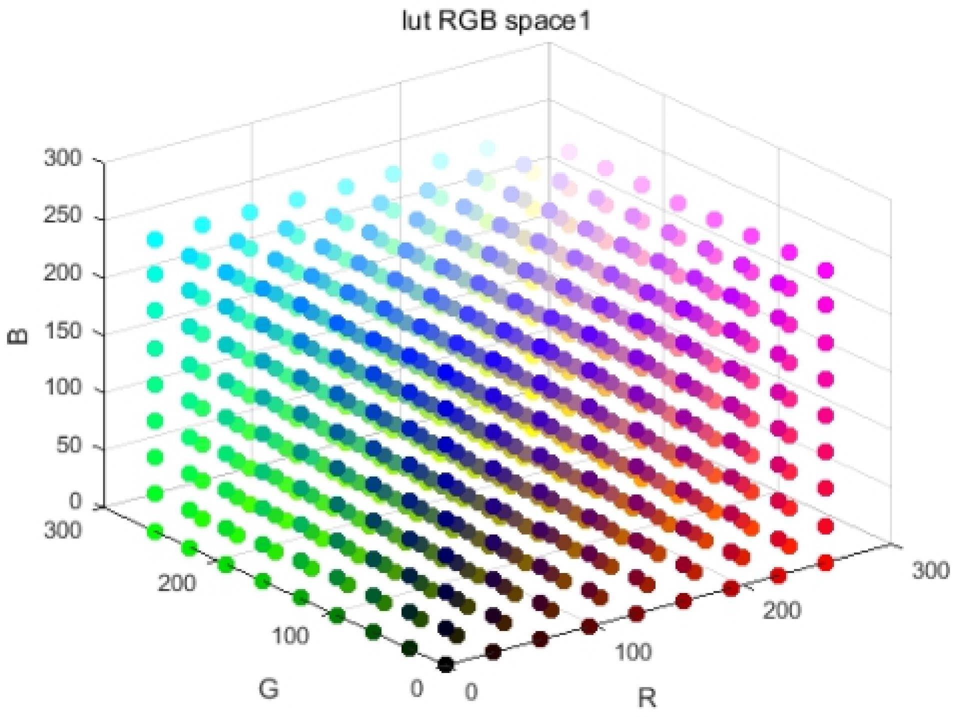
4.2 Function description

3 dimensional look-up-tables (3DLUT)



9x9x9 bypass 3dlut

Any color can be independently mapped to another value



9x9x9 green enhanced 3dlut indication

The 3D LUT on RK1109 is 9x9x9, the values not recorded in the table can be obtained by trilinear interpolation

4.3 Adjustment example

The tool for RK to adjust this function has not yet been developed. The following is a comparison of adjustment effects


之前
之后