



Project of System-on-chip Design

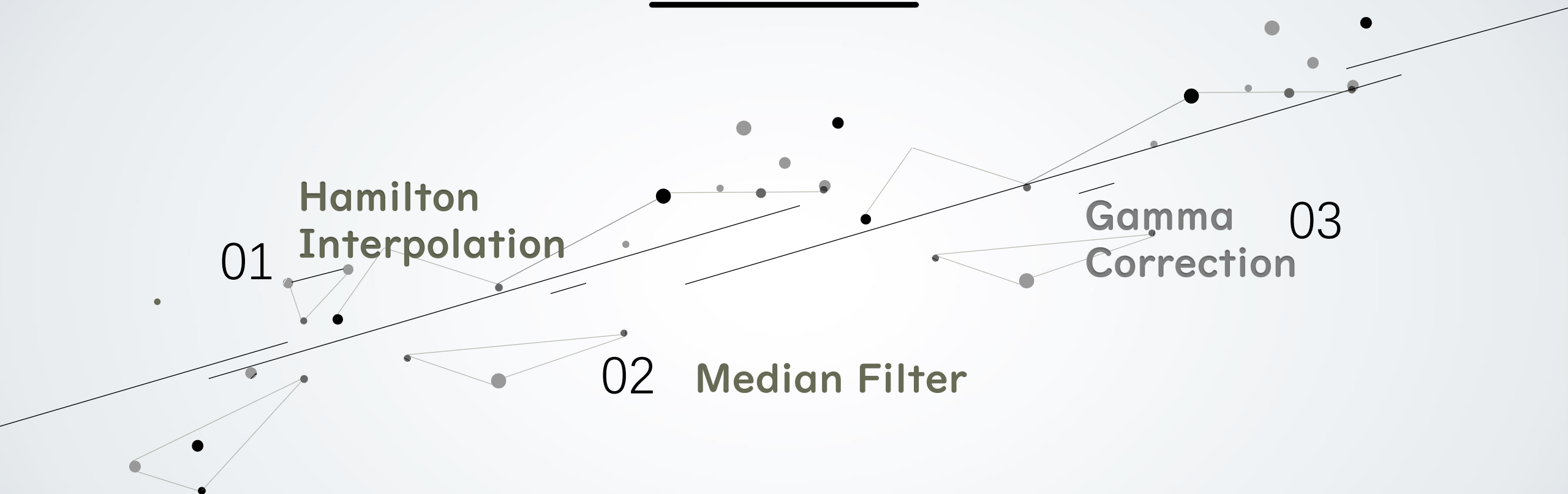
Reporter :
Chen Renwei
Feng Shuo
Wang Haoyu

CONTENT

01 Hamilton Interpolation

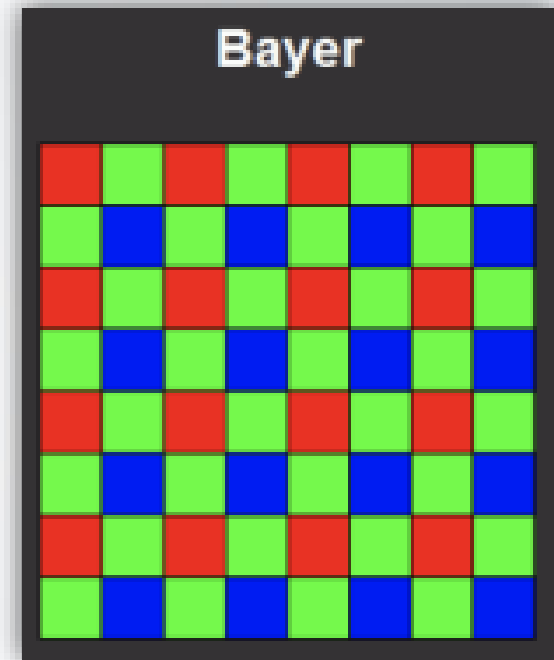
02 Median Filter

Gamma Correction 03



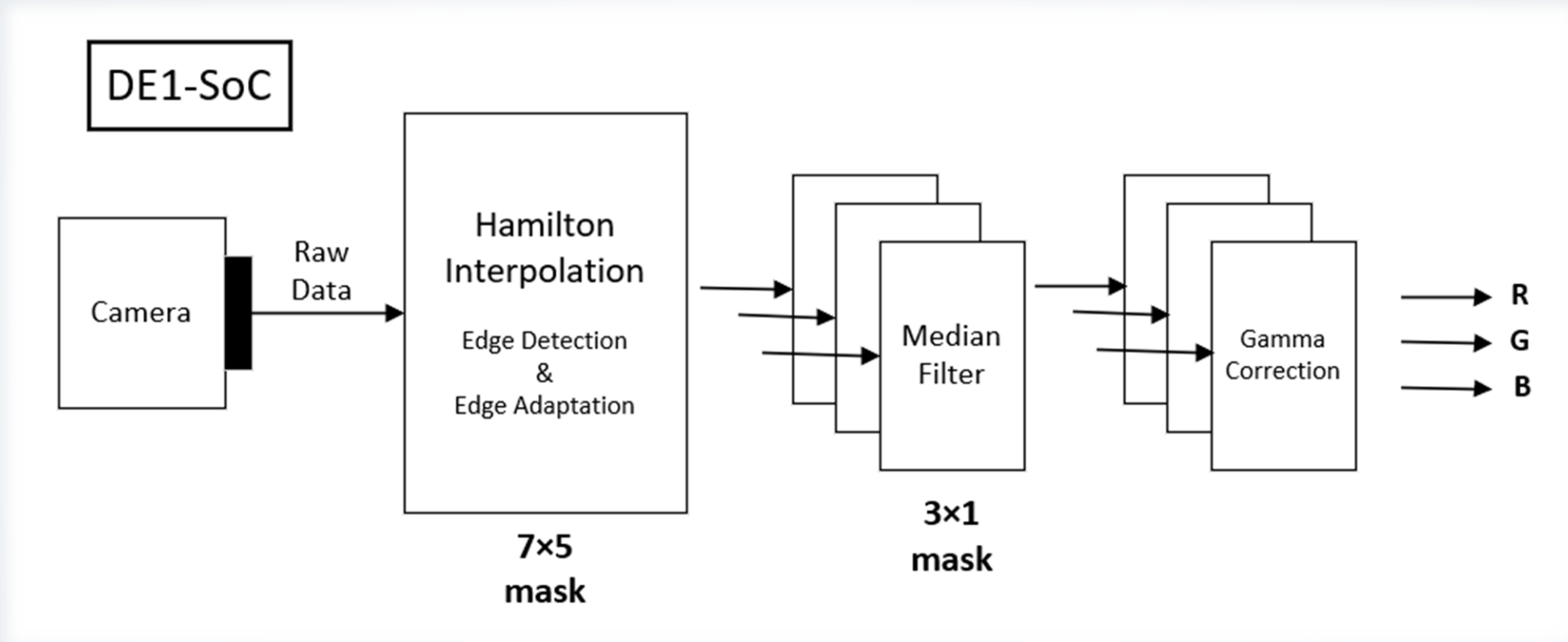
Background of Demosaic

- 图像传感器只能感受光的强弱，无法感受光的波长，故只能拍摄黑白照片。
- 通过在黑白图像的基础上增加彩色滤波结构和彩色信息处理模块，可以获得图像的彩色信息，目前最常用的滤波阵列是棋盘格式的 Bayer CFA，也是本项目所使用的。
- 每个像素点丢失的其他两种颜色分量需要通过后期插值 来重建以获得完整的全彩色图像该插值重建的过程就叫做去马赛克 (Demosaic)



Bayer CFA

Process of Demosaic



- Hamilton Interpolation
- Median Filtering
- Gamma Correction



1

Hamilton Interpolation



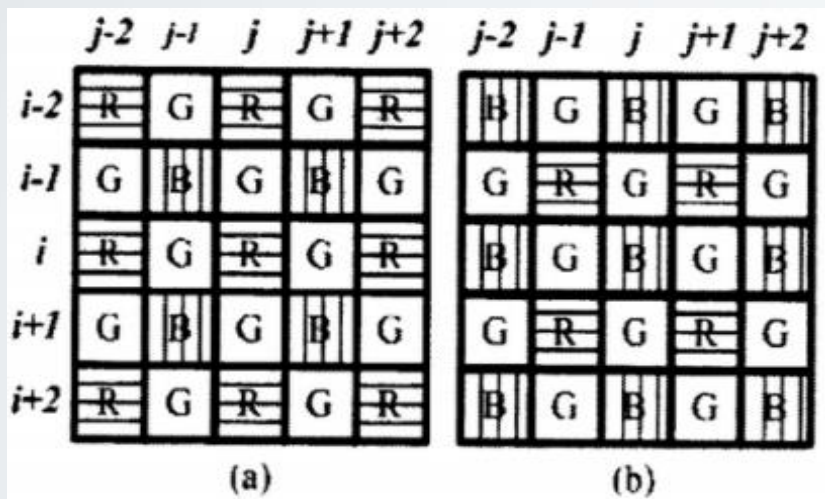
01 / Principle of Hamilton Interpolation

- 红色和蓝色采样点处的绿色分量的重建
- 绿色采样点处的红色和蓝色分量重建
- 红色(蓝色)采样点处的蓝色(红色)分量的重建

优点：绿色分量的插值引入了边缘检测，更好的保留边缘信息；红蓝分量的插值引入梯度信息；利用了颜色空间之间的相关性



01 / 红色和蓝色采样点处绿色分量重建



以图(a)为例

$$\Delta H_{i,j} = |G_{i,j-1} - G_{i,j+1}| + |2R_{i,j} - R_{i,j-2} - R_{i,j+2}|$$

$$\Delta V_{i,j} = |G_{i-1,j} - G_{i+1,j}| + |2R_{i,j} - R_{i-2,j} - R_{i+2,j}|$$

水平/竖直方向检测算子

当水平算子小于垂直算子时，中心点R在水平边缘的概率较大，中心绿色分量的计算沿水平方向进行

$$g_{i,j} = \frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i,j-2} - R_{i,j+2}}{4}$$

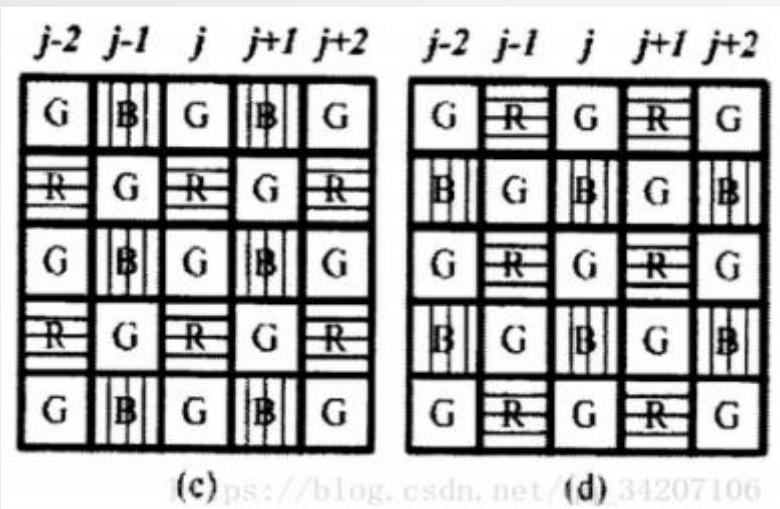
当水平算子大于垂直算子时，中心点R在垂直边缘的概率较大，中心绿色分量的计算沿垂直方向进行

$$g_{i,j} = \frac{G_{i-1,j} + G_{i+1,j}}{2} + \frac{2R_{i,j} - R_{i-2,j} - R_{i+2,j}}{4}$$

水平和垂直的算子相等，则中心点处的绿色分量的计算为水平和垂直方向的平均值

$$g_{i,j} = \frac{G_{i-1,j} + G_{i+1,j} + G_{i,j-1} + G_{i,j+1}}{4} + \frac{4R_{i,j} - R_{i-2,j} - R_{i+2,j} - R_{i,j-2} - R_{i,j+2}}{8}$$

01 / 绿色采样点处的红色和蓝色分量重建



以图(c)为例

中心点处的蓝色分量的重建使用左右两点的B-G空间的线性插值，红色分量的重建使用上下两点的R-G空间的线性插值

$$r_{i,j} = \frac{1}{2}(R_{i-1,j} + R_{i+1,j}) + \frac{1}{2}(2G_{i,j} - g_{i-1,j} - g_{i+1,j})$$

$$b_{i,j} = \frac{1}{2}(B_{i,j-1} + B_{i,j+1}) + \frac{1}{2}(2G_{i,j} - g_{i,j-1} - g_{i,j+1})$$

$$\Delta H_{i,j} = |G_{i,j-1} - G_{i,j+1}| + |2R_{i,j} - R_{i,j-2} - R_{i,j+2}|$$

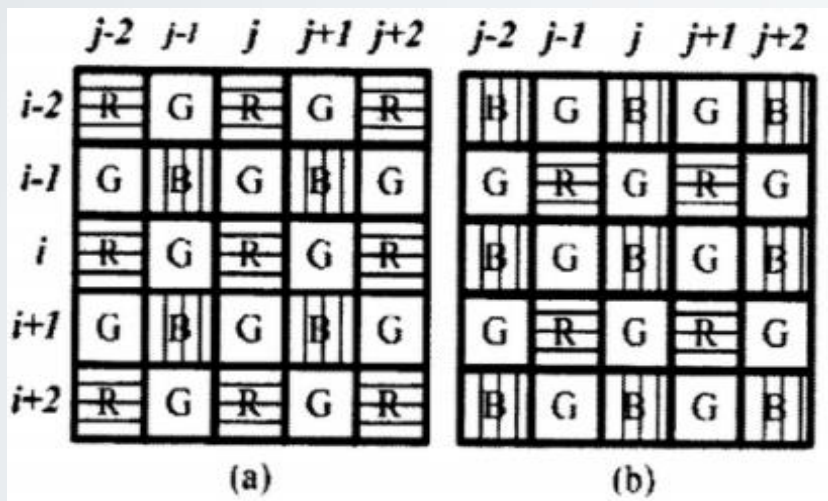
$$\Delta V_{i,j} = |G_{i-1,j} - G_{i+1,j}| + |2R_{i,j} - R_{i-2,j} - R_{i+2,j}|$$

水平/竖直方向检测算子

考虑资源占用情况，故采用7×5，对于非中心的绿色分量采用上下左右取均值的方法提取

01

红色(蓝色)采样点处的蓝色(红色)分量的重建



以图(a)为例

根据梯度的比较结果，选择合适的插值防线

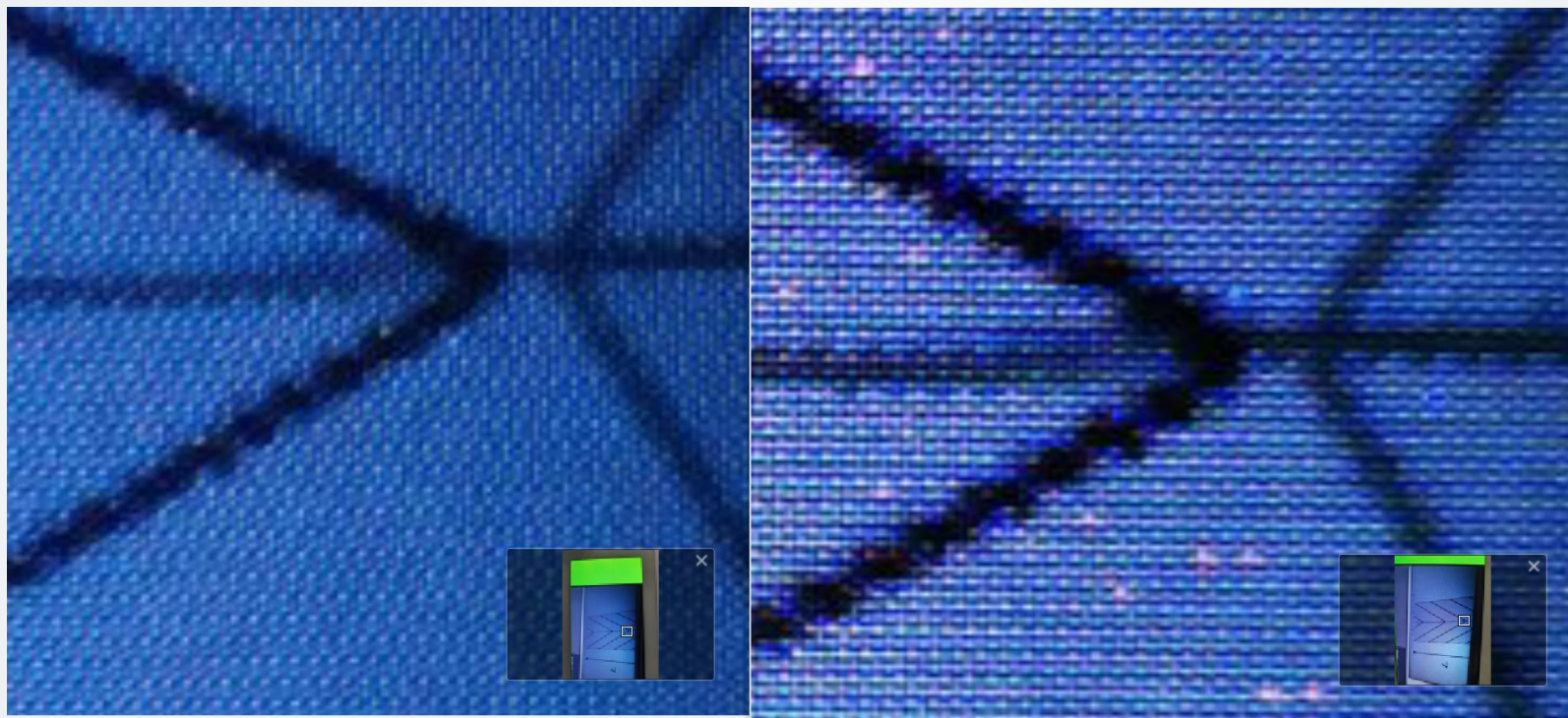
$$D_{45}(i, j) = |B_{i-1, j+1} - B_{i+1, j-1}| + |2g_{i, j} - g_{i-1, j+1} - g_{i+1, j-1}|$$

$$D_{135}(i, j) = |B_{i-1, j-1} - B_{i+1, j+1}| + |2g_{i, j} - g_{i-1, j-1} - g_{i+1, j+1}|$$

左下右上/左上右下梯度计算

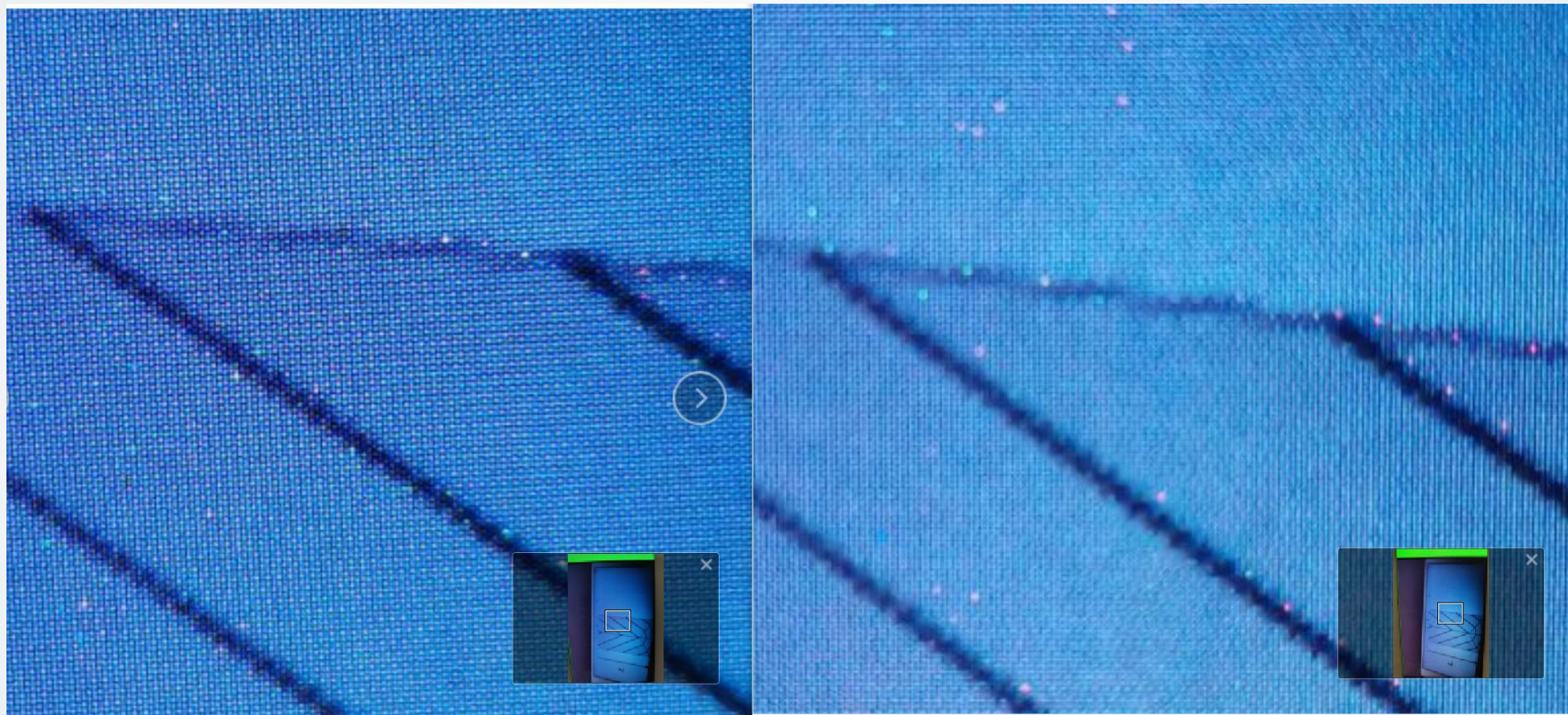
$$r_{i, j} = \begin{cases} \frac{B_{i-1, j+1} + B_{i+1, j-1}}{2} + \frac{2g_{i, j} - g_{i-1, j+1} - g_{i+1, j-1}}{2}, & \text{当 } D_{45} < D_{135} \\ \frac{B_{i-1, j-1} + B_{i+1, j+1}}{2} + \frac{2g_{i, j} - g_{i-1, j-1} - g_{i+1, j+1}}{2}, & \text{当 } D_{135} < D_{45} \\ \frac{B_{i-1, j+1} + B_{i+1, j-1} + B_{i-1, j-1} + B_{i+1, j+1}}{4} + \frac{4g_{i, j} - g_{i-1, j+1} - g_{i+1, j-1} - g_{i-1, j-1} - g_{i+1, j+1}}{4}, & \text{当 } D_{45} = D_{135} \end{cases}$$

01 / 加入汉密尔顿插值算法前后效果对比



左图：优化后；右图：优化前
可以看出左图优化后线条更直，锯齿状效应得到优化，边缘信息更清晰

01 / 閾值設置



左图：最终阈值设定为12，左边的线条相对来说更清晰；
右图：对比图，线条较模糊，不是最佳阈值



02 / 滤波的选择：中值滤波 VS. 均值滤波

图像滤波：

即在尽量保留图像细节特征的前提下对噪声进行抑制，通过抑制高频段来减少噪音，同时会造成图像一定程度上的模糊，这也叫做平滑或者低通滤波器。

均值滤波

均值滤波是空间域**线性滤波器**，其中所得到的图像中的每个像素具有的值等于其邻近的像素的输入图像中的平均值。它是低通（“模糊”）滤波器的一种形式。

中值滤波

中值滤波是一种**非线性滤波器**，利用采集领域的中值的方法，对噪声进行抑制。尤其对于椒盐噪声有用。

02 / 滤波的选择：中值滤波 VS. 均值滤波

均值滤波

均值滤波是空间域**线性滤波器**，它是低通（“模糊”）滤波器的一种形式。

中值滤波

中值滤波是一种**非线性滤波器**，利用采集领域的中值的方法，对噪声进行抑制。尤其对于椒盐噪声有用。



均值滤波处理后的图像



中值滤波处理后的图像

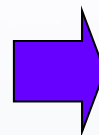
02 / 均值滤波处理方法

以模块运算系数表示即： $H_0 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

待处理像素

1	2	1	4	3
1	2	2	3	4
5	7	6	8	9
5	7	6	8	8
5	6	7	8	9

原图像



1	2	1	4	3
1	3	4	4	4
5	4	5	6	9
5	6	7	8	8
5	6	7	8	9

均值滤波处理后

02 / 均值滤波效果



有噪声的原图像



均值滤波处理后的图像

均值滤波器的缺点是，会使图像变**模糊**，原因是它对所有的点都是同等对待，在将噪声点分摊的同时，将景物的边界点也分摊了。

02 / 中值滤波思路

1	2	1	4	3
1	2	2	3	4
5	7	6	8	9
5	7	6	8	8
5	6	7	8	9

原像素块



1	2	1	4	3
1	2	3	4	4
5	5	6	6	9
5	6	7	8	8
5	6	7	8	9

中值滤波后的像素块

中值滤波：

某些噪声（如椒盐噪声）的像素点比周围的像素亮（暗）许多。

如果在某个模板中，对像素进行由小到大的排列，最亮或者最暗的点（噪声）会被排在两侧。

取模板中排在中间位置上的像素的灰度值替代待处理像素的值，就可以达到滤除噪声的目的。

02 / 滤波的选择：中值滤波效果



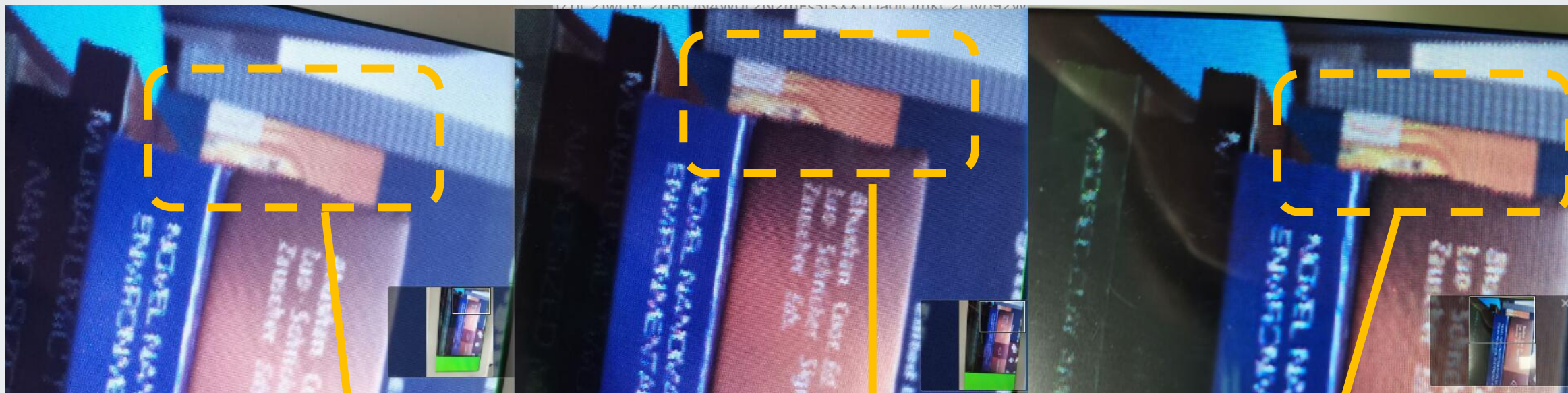
有噪声的原图像



均值滤波处理后的图像

对于椒盐噪声，中值滤波效果比均值滤波效果好。椒盐噪声是幅值近似相等但随机分布在不同位置上，图像中有干净点也有污染点。均值将改变干净点的值。

02 / 中值滤波器效果

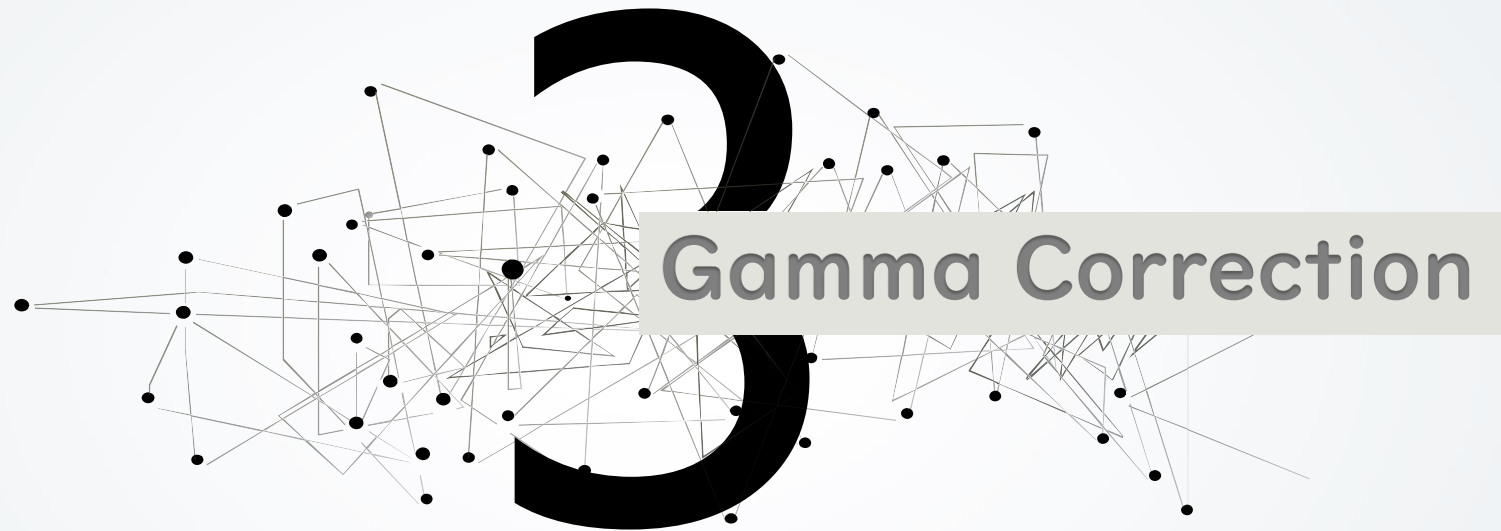


(a) 3*3 中值滤波

(b) 1*3 中值滤波

(c) 无滤波





Gamma Correction



03

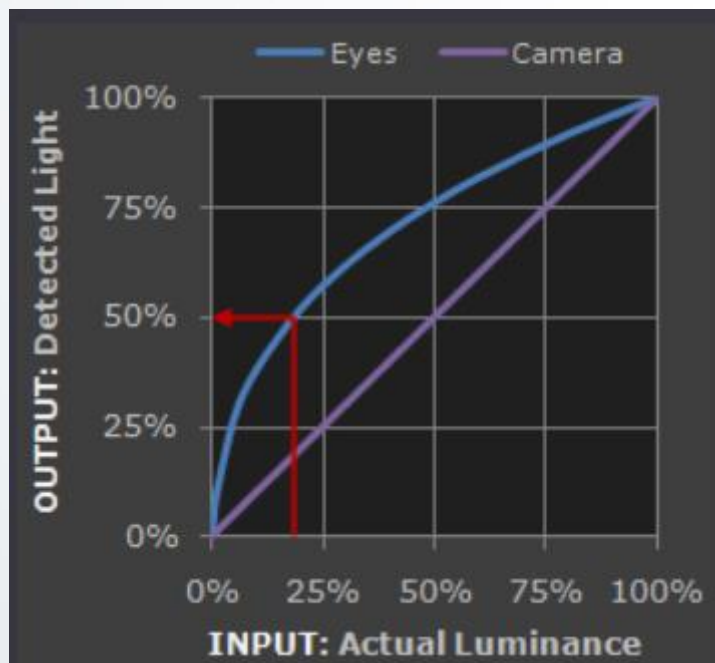
人眼对图像的非线性处理——非线性增长

数码相机：当两倍的光子击中传感器时，它接收到两倍的信号
(一种“线性”关系)

人眼：我们认为两倍光的亮度只是提升了一小部分
(一种“非线性”关系)。

$$S = aI^{1/\gamma}, \gamma > 1$$

S为人眼所感受的光强，I为实际光强



人眼感光与真实光强度的对比

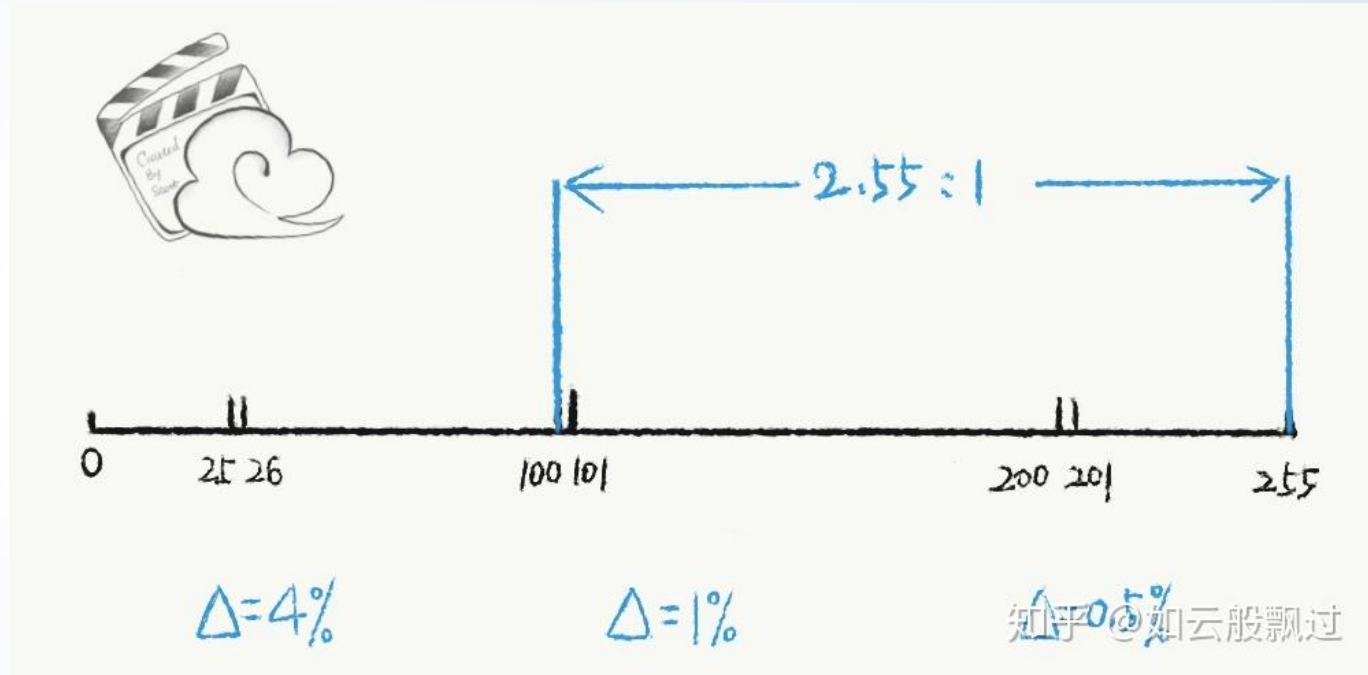
与相机相比，我们对暗色调的变化比对亮色调的变化更敏感，如此我们的视觉在更大的亮度范围内运行。

03

人眼对图像的非线性处理——非线性增长

人眼对亮度变化的分辨率是有限的，在均匀的背景，当前景亮度变化达到背景亮度的1%时，人眼才能感受到亮度的变化

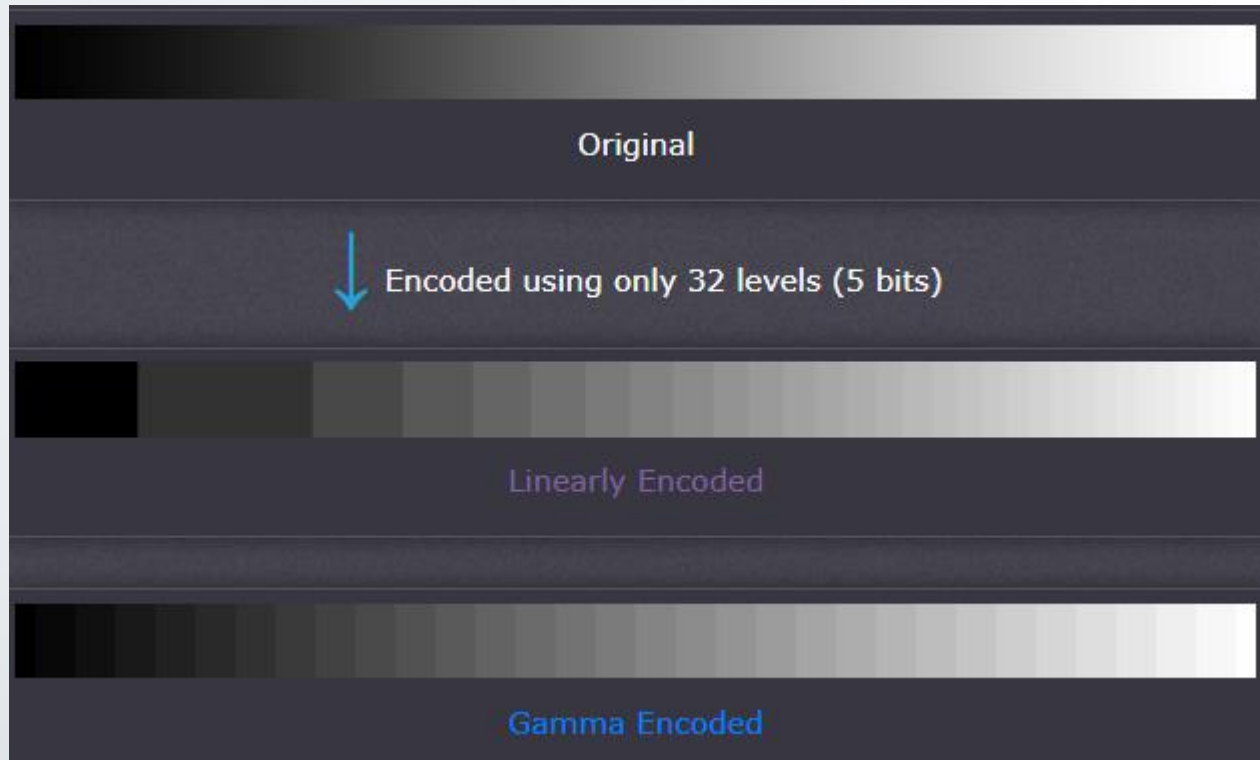
8bit 线性编码，100 以下的编码值每增加 1，其亮度变化必定大于 1%，不能使人感受到平滑过度；只有 100 以上的编码值才符合人眼的特性



03

图像Gamma编码矫正

对摄像机保存的原始图像Gamma编码，因此文件中光强的两倍值更接近于我们所感知到的两倍亮度。



$$V_{out} = aV_{in}^{1/\gamma}, \gamma > 1$$

伽玛编码梯度分布的音调大致均匀地横跨整个范围，而线性编码的过度相对不平滑

03

电子显像管(CRT)的Gamma校正

电子显像管通过控制电流大小来控制显示屏幕上的亮度，然而亮度和电流之间的关系并非线性的：

$$V_{display} = V_{code}^{\gamma}$$

对摄像机保存图像进行一个Gamma校正： $V_{code} = V_{in}^{1/\gamma}$

输出图像： $V_{display} = (V_{in}^{1/\gamma})^{\gamma} = V_{in}$

SRGB标准中，默认显示设备的 γ 值为2.2



03

使用LUT查找表在De1_soc中实现gamma校正

利用matlab生成 $1/\gamma=0.42$ 的查找表mif文件

Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	0	21	28	34	39	43	46	50	..."+.2
8	53	56	58	61	64	66	68	70	58:=@BDF
16	72	74	76	78	80	82	84	85	HJLNPRTU
24	87	89	90	92	93	95	96	98	WYZ_]`b
32	99	101	102	103	105	106	107	109	cefgijkm
40	110	111	112	114	115	116	117	118	noprstuv
48	119	120	122	123	124	125	126	127	wxz{[]~.
56	128	129	130	131	132	133	134	135
64	136	137	138	139	140	141	142	143
72	143	144	145	146	147	148	149	150
80	151	151	152	153	154	155	156	156
88	157	158	159	160	160	161	162	163
96	164	164	165	166	167	167	168	169
104	170	170	171	172	173	173	174	175
112	175	176	177	178	178	179	180	180
120	181	182	182	183	184	184	185	186
128	186	187	188	188	189	190	190	191

03

gamma校正对比



Linear RAW Image
(image gamma = 1.0)



Gamma Encoded Image
(image gamma = 1/2.2)

03

gamma校正对比



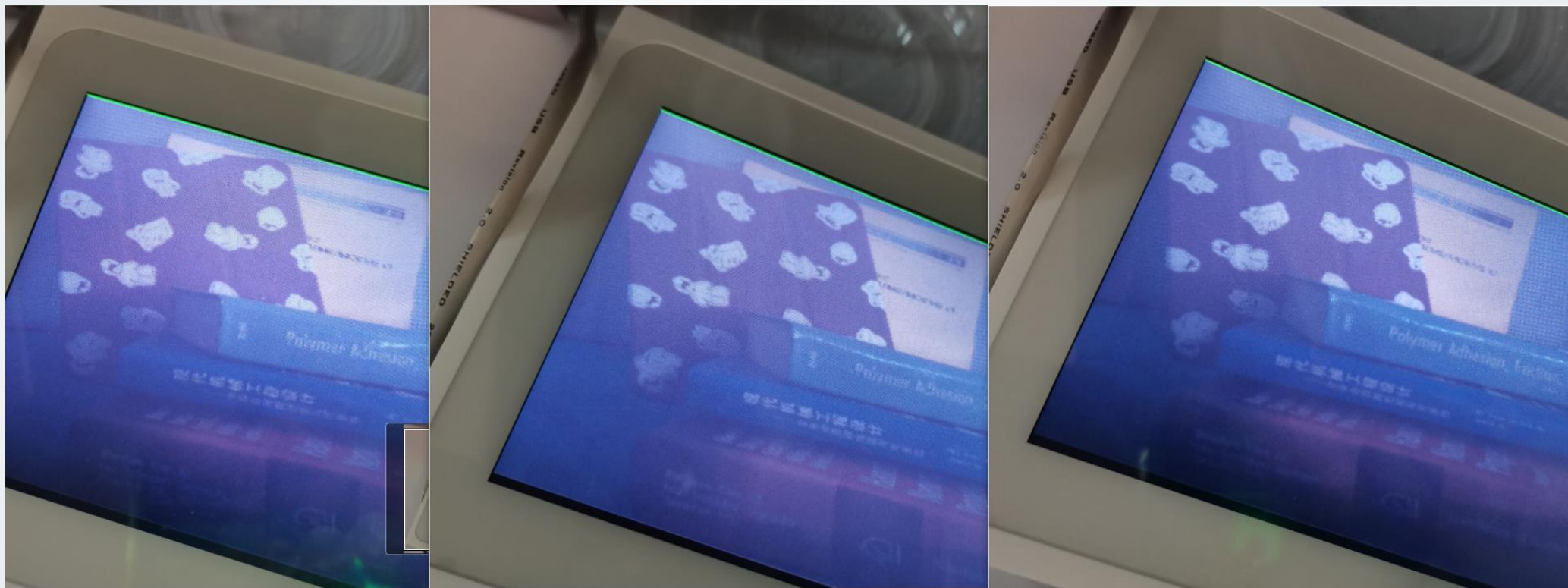
$1/\gamma=0.42$

无gamma校正

03

gamma校正对比

经过对比，确认 $1/\gamma = 0.42$ 时效果最佳



$1/\gamma = 0.25$ (1/4) ; 0.42 (1/2.38) ; 0.75 (1/1.3) ;
从左往右，从亮变暗

04/

Q

&

A



Thank You