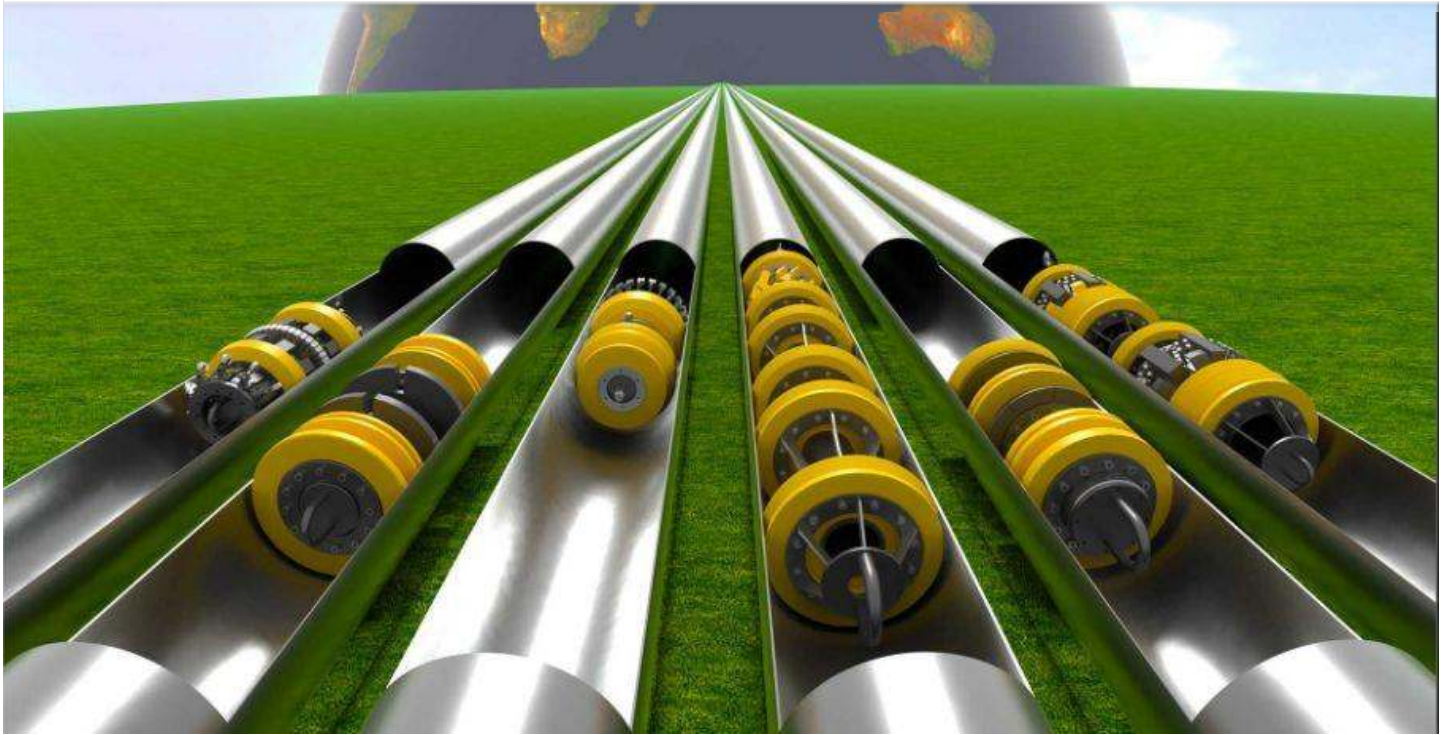


知乎

首發於
無法抗拒的圖像技術

Understanding ISP Pipeline



劉斯寧

Camera技術專家

196 人贊同了該文章

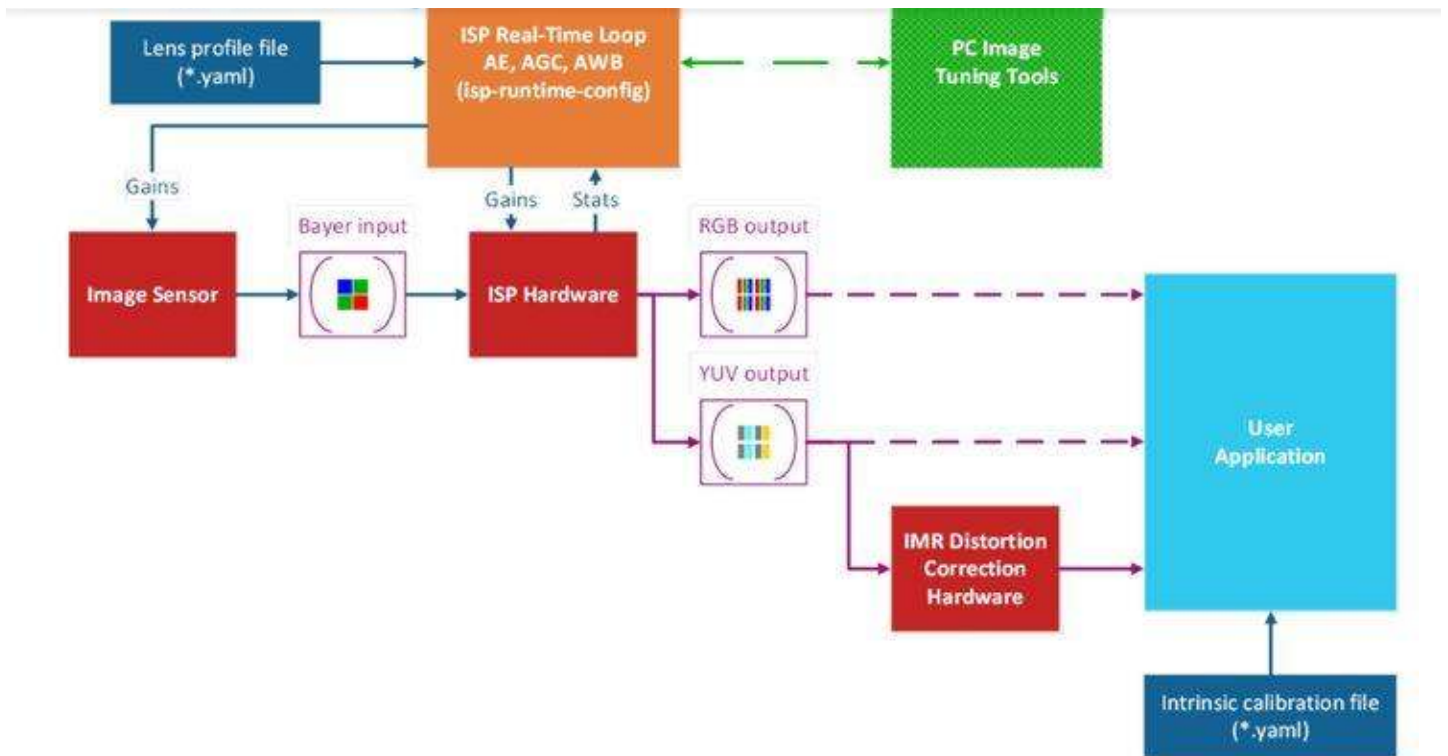
什麼是ISP

主流的CMOS和CCD sensor幾乎都是輸出Bayer mosaic格式的RAW數據，這種數據格式是無法直接觀看的，必須轉換成常見的RGB或YUV格式才能被主流的圖像處理軟件支持。對於camera產品而言，一般還需要將RGB或YUV圖像進一步轉換成JPEG格式以方便進行存儲。上述圖像處理過程統稱圖像信號處理（Image Signal Processing，ISP），廣義的ISP包含了JPEG和H.264/265圖像壓縮處理，而狹義的ISP僅包括從RAW格式變換到RGB或YUV的處理過程。

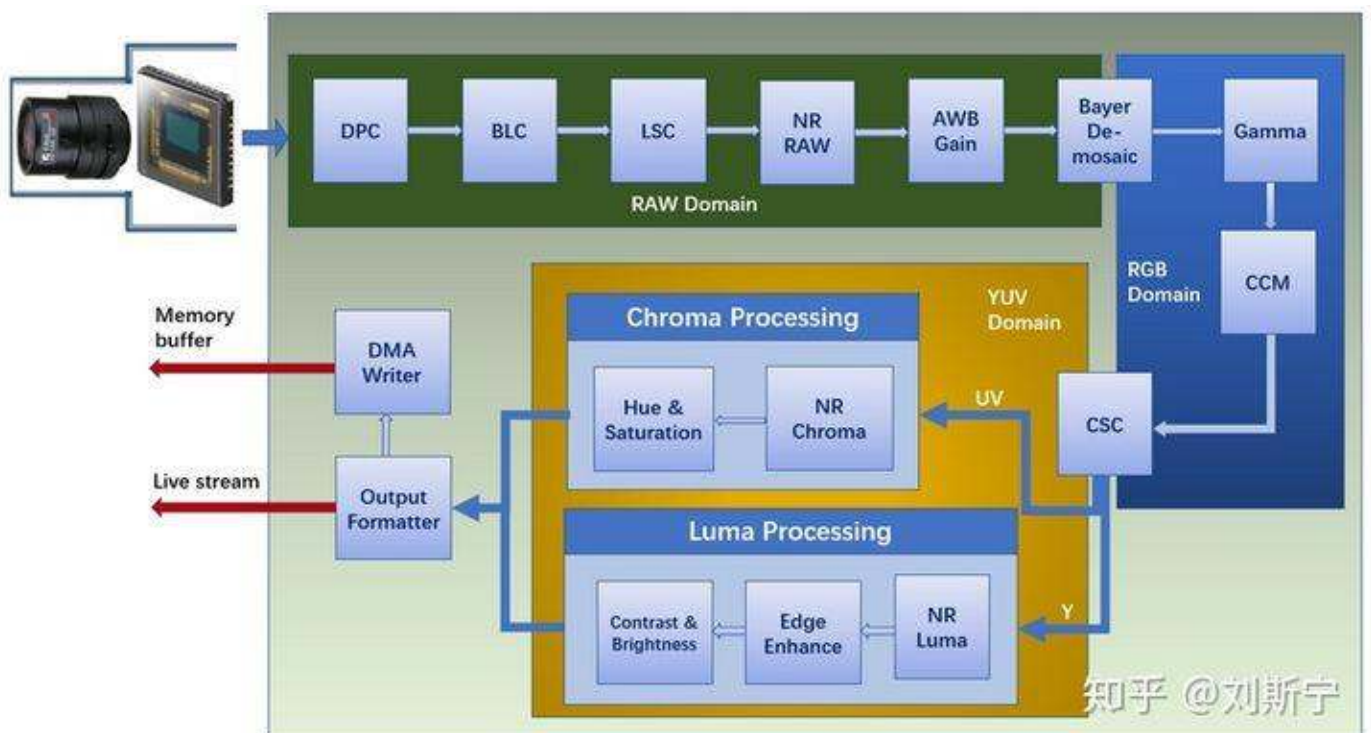
由於圖像信號處理涉及大量的數據和嚴格的實時性要求，所以ISP通常必須採用硬件方案實現。有些Image sensor自身支持一定的ISP功能，用戶可以選擇啟用或者關閉。有些ISP是作為獨立的芯片或者SoC IP產品可以從供應商那裡採購。下圖顯示了一個典型camera系統的功能框圖，主要包括圖像傳感器（Image Sensor）、ISP硬件（ISP Hardware）、ISP軟件（ISP Real-Time Loop）等核心單元，以及ISP調試工具（PC Image Tuning Tools）、用戶程序（User Application）、配置文件等。有些ISP硬件會內嵌支持一些CV算法功能，如鏡頭畸變校正（Distortion Correction）。



知乎

首發於
無法抗拒的圖像技術

一個典型的ISP流水線由一系列處理模塊組成，這些模塊首尾相連，在幾百MHz的時鐘驅動下同時高速運轉，圖像數據不斷從一個模塊轉移至下一個模塊，直到完成所有的算法處理，最終以YUV或RGB的形式從流水線的末級流出ISP。下圖所示的是一個支持常見基本功能的ISP流水線。



從圖中可以看到，圖像數據在ISP內部經歷了兩次顏色空間變換，第一次變換發生在Demosaic模塊，它把像素從RAW域變換到RGB域，第二次變換發生在CSC模塊，它把像素從RGB變到YUV域。下表對ISP 各模塊的作用給予了簡要說明。

知乎

首發於
無法抗拒的圖像技術

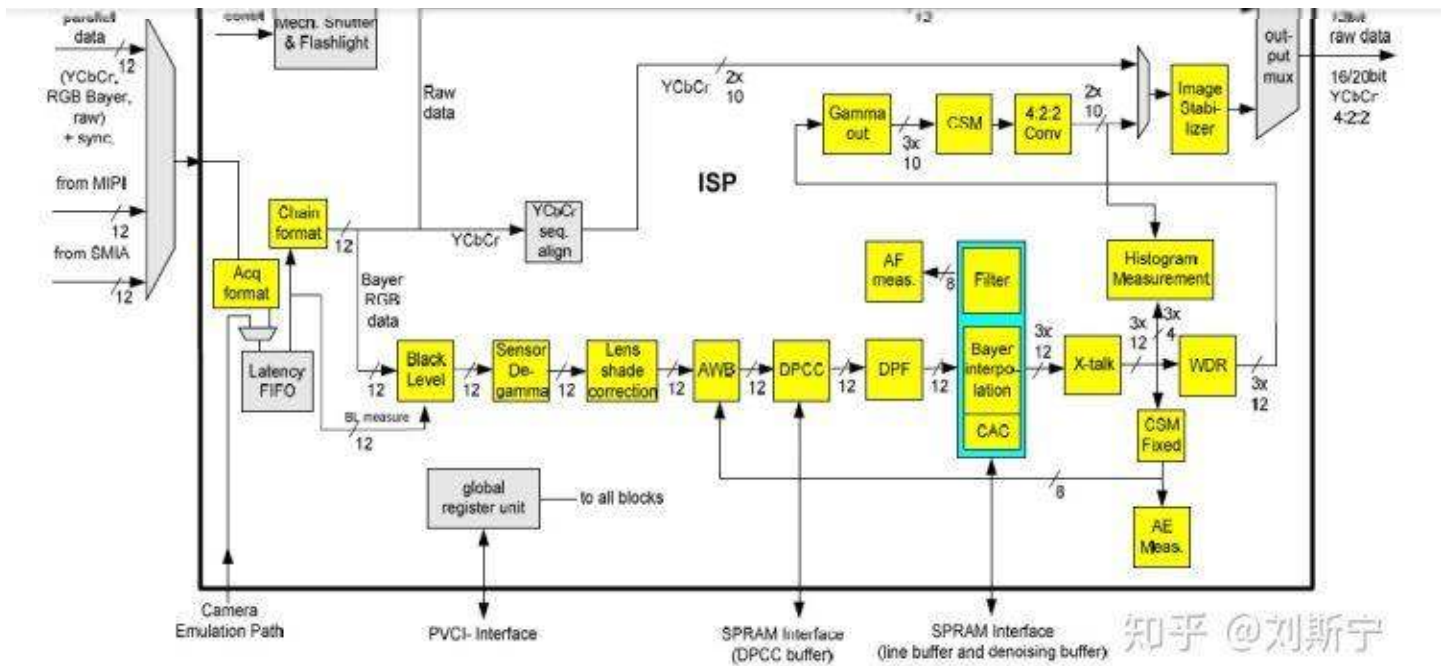
1	DPC	Defective Pixel Concealment	坏点校正	
2	BLC	Black Level Correction	黑电平校正	
3	LSC	Lens Shading Correction	镜头阴影校正	
4	NR RAW	Noise Reduction for RAW	RAW域降噪	
5	AWB Gain	Auto White Balance Gain	白平衡增益	
6	Bayer Demosaic	Bayer Demosaic	RGB插值	
7	Gamma	Gamma	Gamma校正	
8	CCM	Color Correction Matrix	颜色矫正矩阵	
9	CSC	Color Space Conversion	颜色空间变换	
10	NR Chroma	Noise Reduction for Chroma	颜色降噪	知乎 @刘斯宁

11	Hue & Saturation	Hue & Saturation Control	色调、色饱和度控制	
12	Edge Enhance	Edge Enhance	边缘增强	
13	Contrast & Brightness	Contrast & Brightness Control	对比度、亮度控制	
14	Data Formatter	Data Formatter	图像格式转换	
15	DMA Writer	DMA Write Controller	写DMA	知乎 @刘斯宁

下面是Silicon Image ISP的功能框圖，這個ISP歷史比較悠久了，主要以IP授權的形式賣給芯片設計公司，在中國也有不少客戶。這個IP最初設計的時候是面向非專業的消費領域，所以只支持一些最基本的功能，對降噪、白平衡、WDR等重點特性的支持十分薄弱，目前已經難以滿足主流需求了。筆者曾在2018年的一個車載流媒體項目中使用某國產廉價sensor搭配這款ISP，堪稱是職業生涯中最慘痛的一段遭遇。



知乎

首發於
無法抗拒的圖像技術

ACQ (Acquisition) , 適配輸入RAW圖的格式

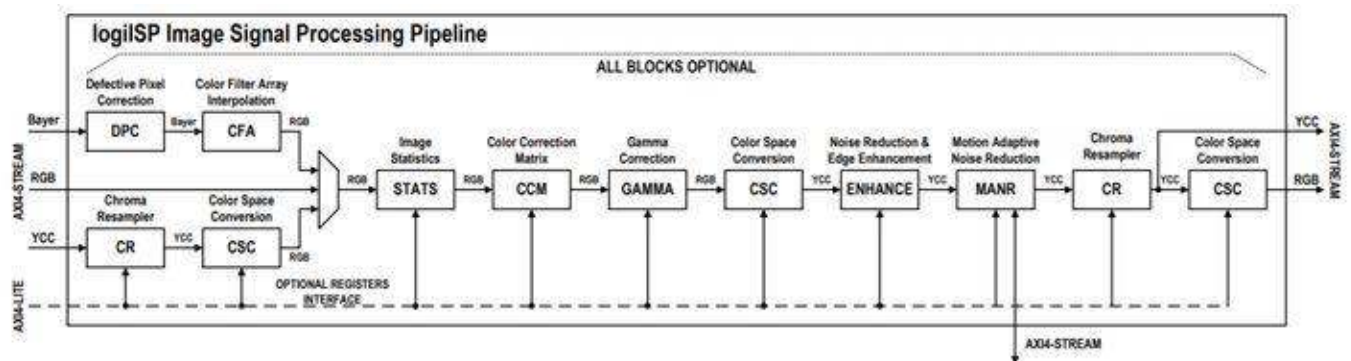
CNR (Chromatic Noise Reduction) , 色度降噪

X-talk (Crosstalk) , 串擾, 即CCM。其實用Crosstalk指代CCM並不準確, 因為它並不是引入CCM的主要原因, 這是一種流傳頗廣的誤解。

VSM (Video Stablization Measurement) 視頻穩定

CSM (Color Space Matrix) , 即CSC(Color Space Conversion) , RGB2YUV 空間變換

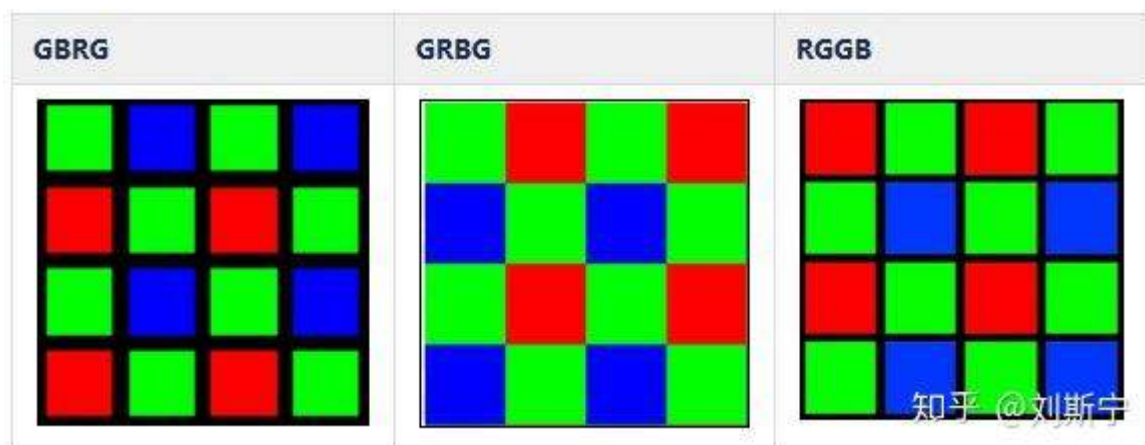
下面是logiISP的功能框圖, 可以配合Xilinx Zync FPGA使用, 提供最簡單的ISP功能。



ISP輸入圖像的格式



常用的Bayer格式有RGGB、GRBG、GBRG等多種，因此需要正確配置ISP以反應sensor的數據格式。



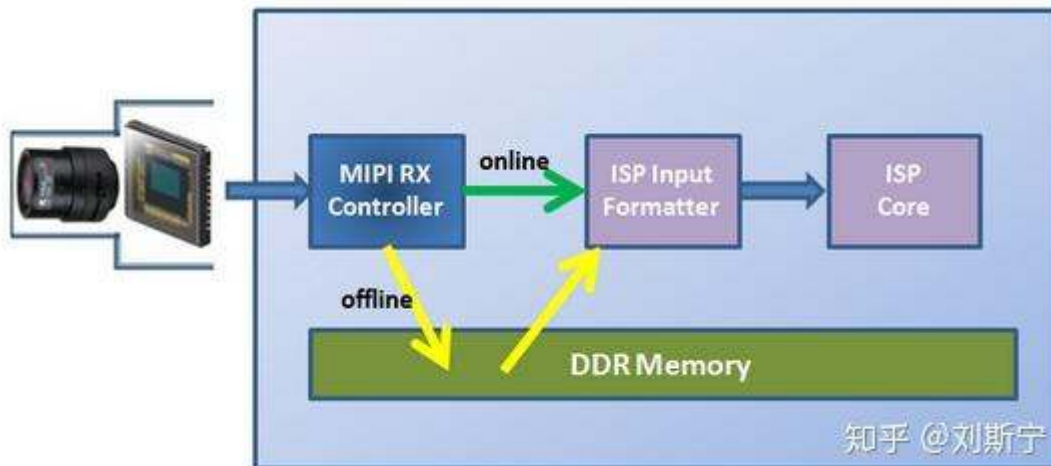
RAW數據的精度常見有8/10/12/14bit等規格。安防監控行業較多使用10/12bit精度的sensor，醫療行業則主要使用12bit以上精度sensor，單反和廣電行業則主要使用14bit精度sensor。當精度高於8bit時，就會自然帶來如何存儲和表示數據的問題。多數sensor支持固定用16bit數據表示10/12/14bit的有效數據，這種表示方法簡單易行，但是在傳輸過程中會浪費一些帶寬，在存儲環節會浪費一些存儲空間。對與一些帶寬和存儲資源特別緊張的場合，有些sensor會支持壓縮表示以節約帶寬，但是這就需要ISP能夠支持相應的壓縮格式，否則就需要增加格式適配處理環節，無論是採用硬件還是軟件方式實現都會增加系統的複雜度和成本。

另外，很多相機、攝像機產品都會在鏡頭光路中插入一個紅外截止濾光片，其作用是允許波長小於截止頻率的光進入系統，而阻止波長大於截止頻率的光。常用的截止頻率有630nm和650nm兩種，它們允許不同程度的紅光進入成像系統，因此會對白平衡和顏色校正算法產生影響，所以ISP的參數配置必須要與實際選用的截止頻率相配置。



使用ISP處理圖像數據時有兩種常用的數據接入方式，即

- 在線模式，online mode，sensor產生的實時數據和時序控制信號以行為單位送入ISP進行處理
- 離線模式，offline mode，待處理的圖像以幀為單位存儲於系統內存，需要處理時由一個控制邏輯通過DMA從內存中讀取數據，並添加模擬sensor行為的時序控制信號，然後送給ISP進行處理。



這兩種方式在ISP本身看來並沒有本質區別，但是從系統角度看，在線模式具備低延遲(low latency)的優點，具體表現是一幀圖像的第一個像素數據流出sensor後馬上就進入ISP流水線開始處理，而在離線模式下，ISP通常需要等到一幀圖像的最後一個像素數據到齊之後才開始啟動處理。某些海思系列的ISP芯片支持一種特殊的“低延遲”離線模式，即一幀圖像的第X行數據到齊之後即啟動ISP進行處理，而不必等到最後一個像素到齊。這種模式自然會引入一個問題，就是如果ISP處理的速度比sensor輸出數據的速度快，就會發生ISP讀指針很快越過sensor寫指針，導致ISP讀取到無效數據。下面舉一個典型的例子，

假設一個1080p的sensor，主頻(master clock frequency)是76MHz，即一個clock cycle是13.15ns。

假設該sensor的配置是每行2000個cycle，這是一個比較典型的配置，其中包含了1920個有效輸出和80個horizontal blanking，則sensor每輸出一行數據將消耗 $13.15\text{ns} \times 2000 = 26.3\mu\text{s}$ ，而輸出全部1080行有效像素則需要 $26.3\mu\text{s} \times 1080 = 28.4\text{ms}$ 。

而一個典型的ISP處理完這幀1080p圖像需要多久呢？只需要3ms！也就是說，ISP只有在26ms之後啟動處理才不至於發生讀指針越過寫指針的情況。這種方法可以簡易地規避ISP讀到無效數據的問題，但是也只能爭取到不足3ms的低延遲收益，延遲性能的提升起始並不明顯，可以說是聊勝於無。因此海思又設計了一個更加積極的辦法，即增加了一個硬件機制可以確保讀指針不會越過寫指針，這樣用戶就可以選擇更早地開始啟動ISP處理，當ISP讀指針追上sensor寫指針時，硬件自動插入延時週期令ISP空轉等待，以保證數據完整性。



的作用。

行緩沖 Line Buffer

不論是在線模式還是離線模式，ISP處理圖像都是以行為單位的，所以ISP模塊都會設計一個line buffer可以緩存若干行圖像。通常這個line buffer的大小就決定了這個ISP所支持的最大分辨率。舉例來說，如果一個ISP的line buffer可以容納每行2048個像素，則它無法支持超過2k/1080p的分辨率規格。

低延遲系統 Low-Latency System

延遲（Latency）是camera的一個重要技術指標，某些應用場景對視頻延遲非常敏感，比如競速無人機應用，參賽選手需要操縱高速飛行的無人機躲避障礙，做出各種特技動作。當無人機時速200km時，每秒會飛行55米，如果參賽選手看到的camera圖像總是0.1秒以前的畫面，這會對選手預判無人機的實際位置帶來不少困難。因此凡是涉及到高速運動的場合都會希望camera輸出的畫面盡可能是實時的。

低延遲是一個系統級特性，它需要camera內部各個子系統都要按照最小延遲的原則做設計。一個基本的原則是，所有針對圖像的處理都必須是以行為單位，而不能以幀為單位。假設H.264編碼器能夠工作的最低要求是16行，這就意味著一幀圖像的第一個像素最快也要等到16行數據全部到齊後才能被編碼。假設CMOS sensor輸出速率為每秒120幀，則每幀圖像8.3ms，輸出每行數據需要7.7us，16行緩存意味著camera的理論延遲最低可以做到123us。如果採用非優化的常規技術，編碼器等一幀圖像到齊後才開始編碼，則第一個像素需要空等8.3ms之後才能開始被編碼器處理，而編碼一幀圖像本身可能需要消耗8ms時間，於是第一個像素實際上需要等待16ms以上才能真正進入信道傳輸環節。與123us的理論極限相比，效率相差100倍以上的。

數據對齊 Alignment

ISP、CODEC等硬件單元在處理圖像時通常都會有粒度（granularity）要求，即必須將8/16/32/64/128個像素作為一組來處理，這樣就可以通過硬件並行化來提高吞吐率。這個需求稱為ISP的數據對齊（alignment）需求，多數sensor都支持一個linesize屬性，以保證sensor輸出的每行數據的寬度符合ISP的對齊要求。與linesize等價的術語還有stride，pitch，均表示每行數據所佔用的實際存儲空間。當linesize與圖像的實際分辨率不相等時，sensor會用一個數據填充對齊部分，這個數據可以是固定值0（zero-padding），也可以是本行最後一個像素的值（copy-padding）。

在使用通用CPU處理圖像時也廣泛存在類似的概念。如Intel架構支持的MMX/SSE指令，ARM架構支持的NEON指令，以及Ceva, Cadence等DSP架構支持的Vector指令，都是基於SIMD（Single Instruction Multiple Data）原理，即單個指令並行處理多個數據，從而獲得幾十倍



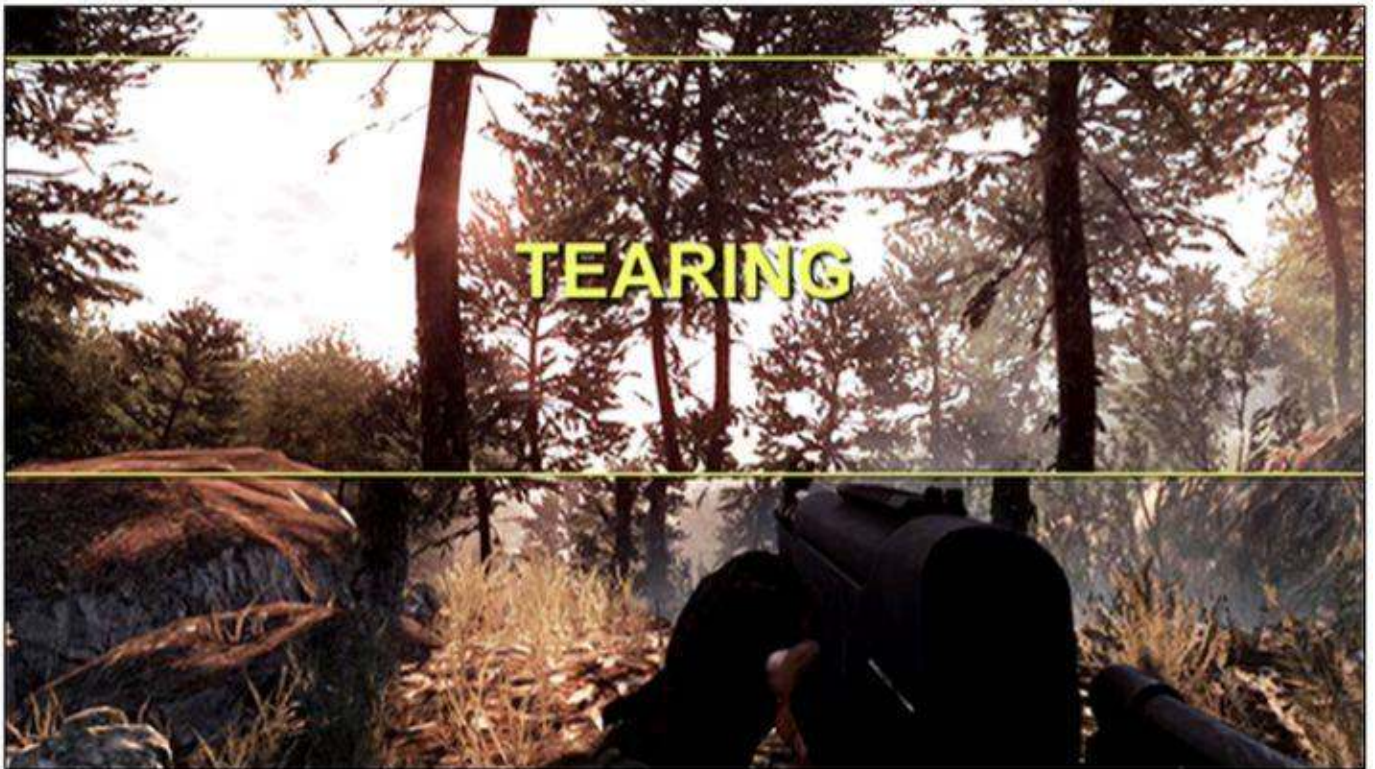
個常見的例子是，某vector指令支持32個操作數，但其中只有前8個是有效數據，輸出buffer也只預留了8個數據的位置，相鄰內存不允許踩踏。盡管DSP在執行指令時必須同時操作32個數據，但計算完成後只能允許前8個數據寫入內存。此時需要將Predicate寄存器的對應bit置1，其餘bit置0，以通知哪些數據是允許保存的。

圖像撕裂 Image Tearing

在離線模式下，從sensor出來的數據需要先進入內存然後再被ISP處理。

在任何模式下，從ISP輸出的數據通常都需要先寫入內存（ring buffer）等待下級模塊處理。

數據寫入內存前，會先進入DMA硬件模塊的緩沖，當DMA模塊申請到總線使用權後才能真正寫入存儲器。於是問題就出現了，當系統非常繁忙的時候，DMA可能長時間未能申請到總線，而sensor數據卻通過專用通道源源不斷地進入DMA輸入緩沖，當DMA緩沖溢出時就會發生數據踩踏事件，未能進入緩沖的數據就丟失了。結果就是圖像撕裂、錯位，即類似下圖所示的情況。

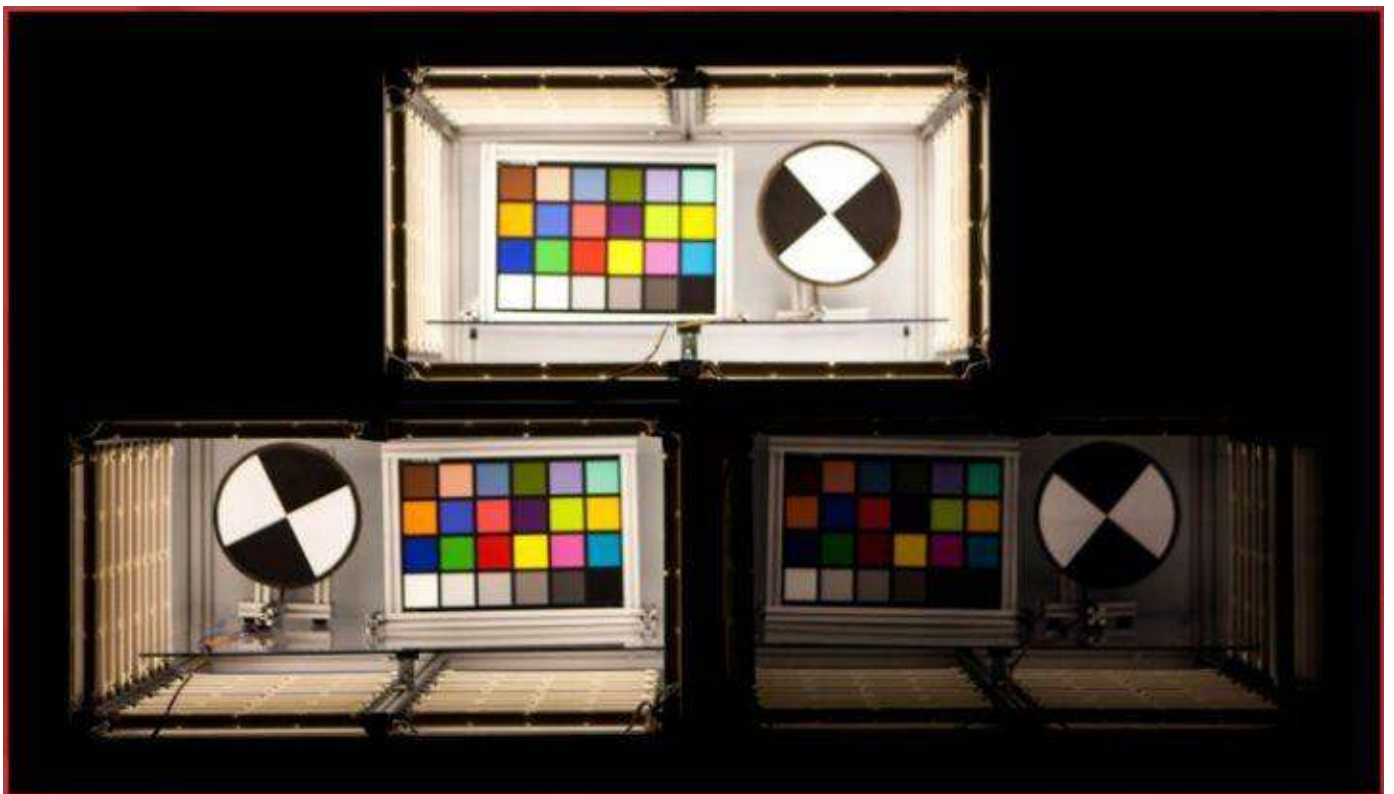


數據踩踏不僅會破壞這一幀圖像，根據系統設計還有可能影響下一幀圖像。有的系統判斷一幀圖像完成的指標是DMA輸出計數，當DMA輸出足夠數量的像素（比如1920x1080）時認為當前圖像傳輸完畢，此時會發出中斷信號。於是問題來了，當已經發生數據踩踏問題時，DMA輸出像素的數量實際上小於應有的數量，於是DMA會一直等到下一幀圖像（N+1）到來，把（N+1）圖像的開始部分計入當前幀才能結束當前幀的工作循環，而下一個工作循環需要從檢測到FrameStart信號開始，結果就是（N+1）圖像的後續部分丟失，直到（N+2）圖像的FrameStart到來後系統才能恢復正常。

負荷的一個簡單辦法是降低輸入幀率或分辨率，也可以考慮在總線仲裁策略上加以調整，對帶寬大戶加以限制，甚至可以在軟件算法上加以調整，將大的操作分解稱若干個小的操作，中間留出一定的空隙給其它任務。這些策略有點像那個買房的笑話，說某國住房資源緊張，房價很高，於是國家鼓勵錯峰買房，有些人這輩子買房，有些人下輩子買房。。。

WDR圖像接入ISP的方式

下圖是在實驗室中評估camera動態範圍的一種常用方法，三個燈箱分別提供低、中、高三種亮度，通過精確控制高亮和低亮的照度比值可以計算出camera的實際動態範圍。



由於材料和工藝的限制，普通的sensor 一般可以提供50~70dB的動態範圍，比較優秀的技術（SONY）可以提供80dB的動態範圍。但是室外應用經常需要捕捉100~110dB的動態範圍，因此人們非常希望sensor 能夠提供120dB以上的動態範圍。筆者在這篇相關作品中分析過，

劉斯寧：Understanding CMOS Image Sensor

zhuanlan.zhihu.com



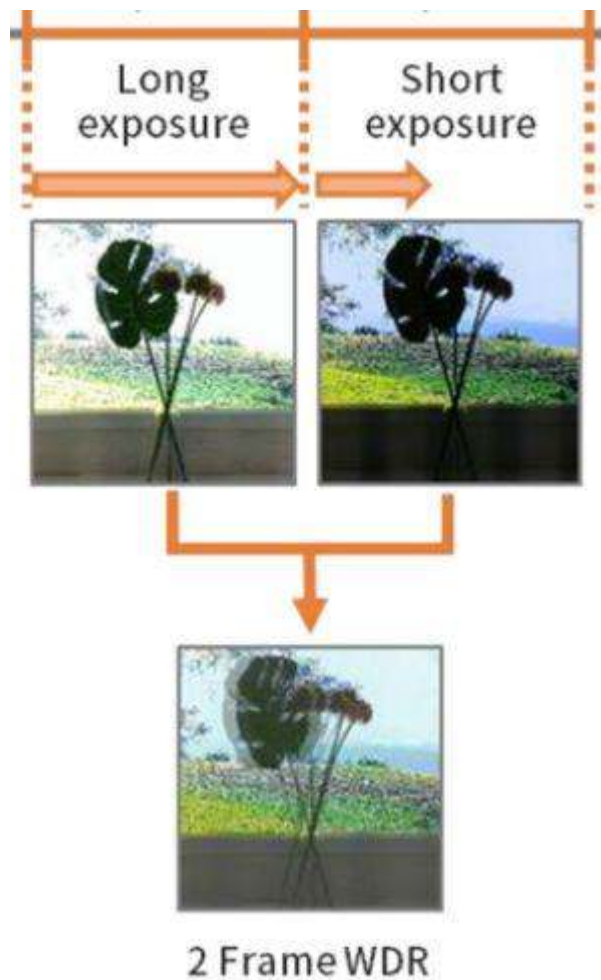
單純增大像素面積以提高sensor動態範圍會遇到成本和技術瓶頸，目前能夠落地的80dB其實已經到達了這條路線的極限。想要進一步提高sensor動態範圍只能另闢蹊徑，於是人們轉向了將多幀不同曝光的圖像融合成一幀寬動態圖像的技術，基本上解決了寬動態的需求。



信息。算法對圖像進行融合的過程稱為frame stitching，或者稱為WDR fusion，意思是將多個圖像融合拼合為一體。

如果希望camera輸出幀率保持30fps不變，則兩幀融合WDR需要sensor輸出60fps，而三幀融合WDR需要sensor輸出90fps，四幀融合WDR需要sensor輸出120fps。這也是為什麼主流產品以兩幀合成WDR為主的原因。

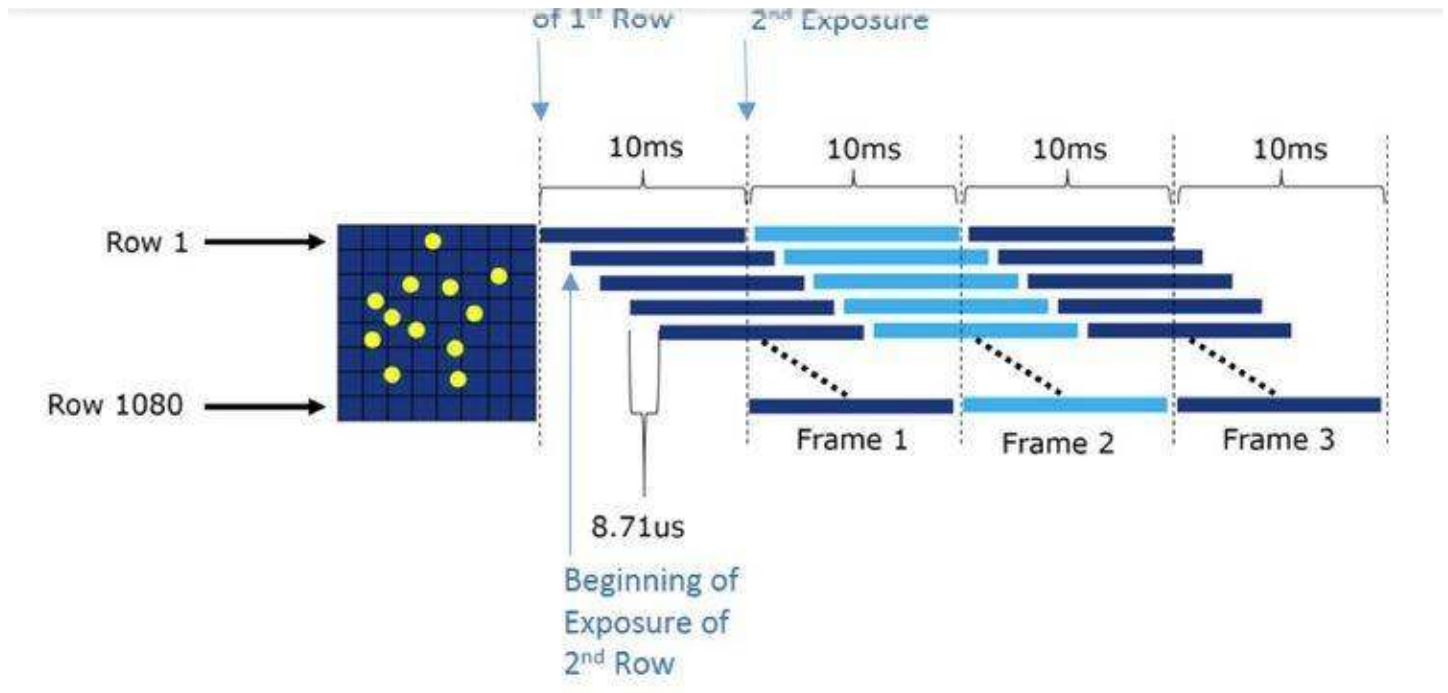
同樣是兩幀合成一幀，也可以有不同的實現方法。傳統的做法是sensor先後輸出兩幀圖像到內存中，兩幀圖像全部就位後再交給ISP進行處理。由於前後兩幀時間的間隔比較大，圖像中如果有運動的物體就會產生一些問題，如下圖所示。這種問題叫做motion artifacts，目前基本沒有太好的處理辦法。



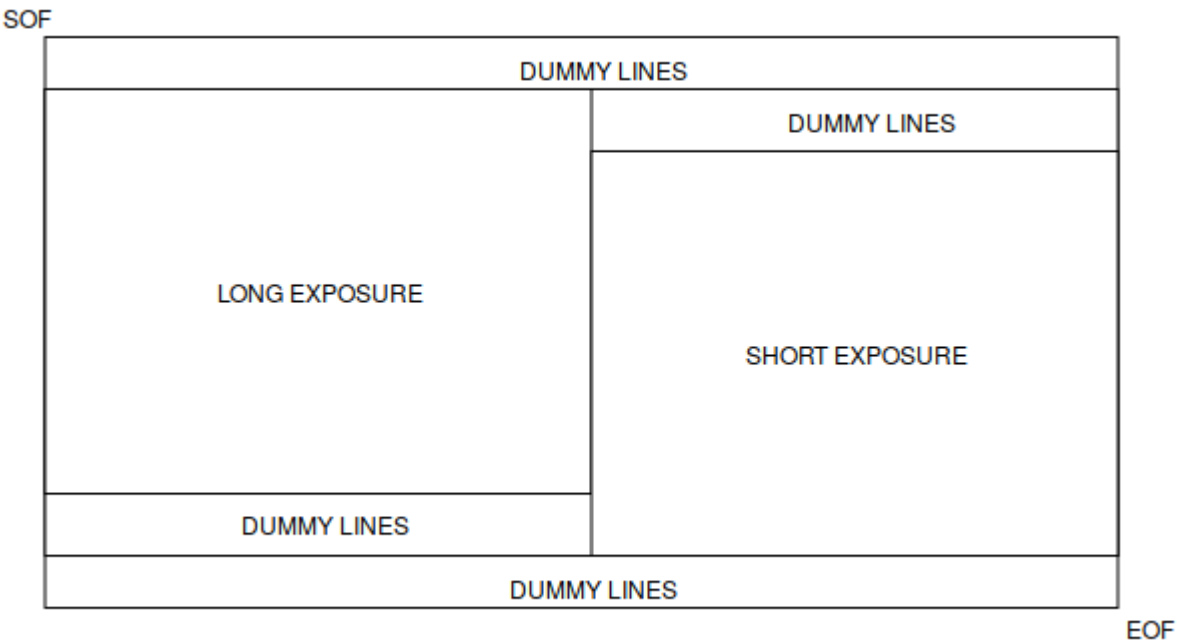
如果sensor 不再以幀為單位輸出，而是以行位單位輸出，則可以緩解運動偽影問題。如下圖所示，sensor 先輸出一行長曝光像素，再輸出一行短曝光像素，然後開始輸出下一行。當最後一個像素掃描完畢時，sensor已經完成了兩幀圖像的輸出。三/四幀合成WDR也是同理。

知乎

首發於
無法抗拒的圖像技術



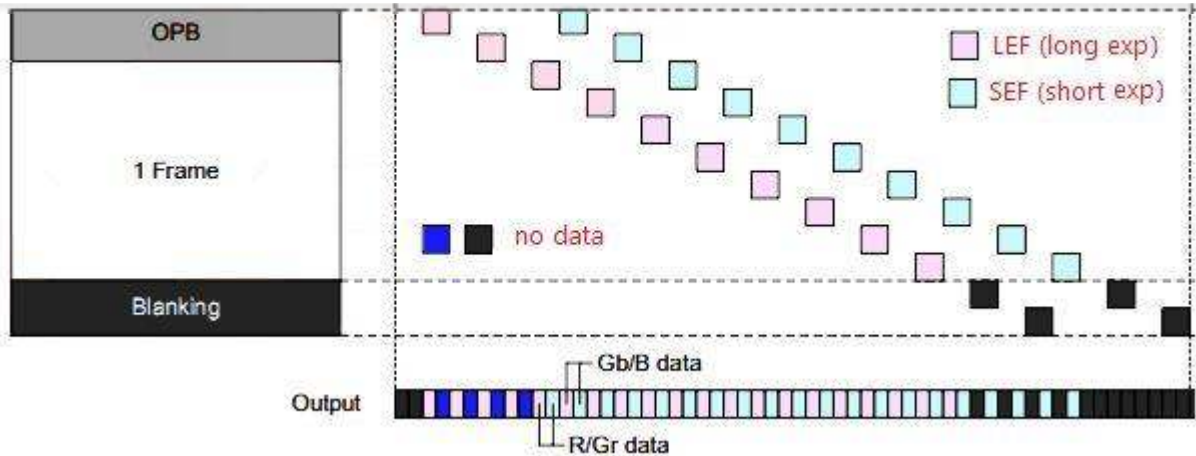
這種輸出方式被OmniVision 稱為Staggered WDR技術，下面是OV Staggered WDR sensor 的 幀結構示意圖，



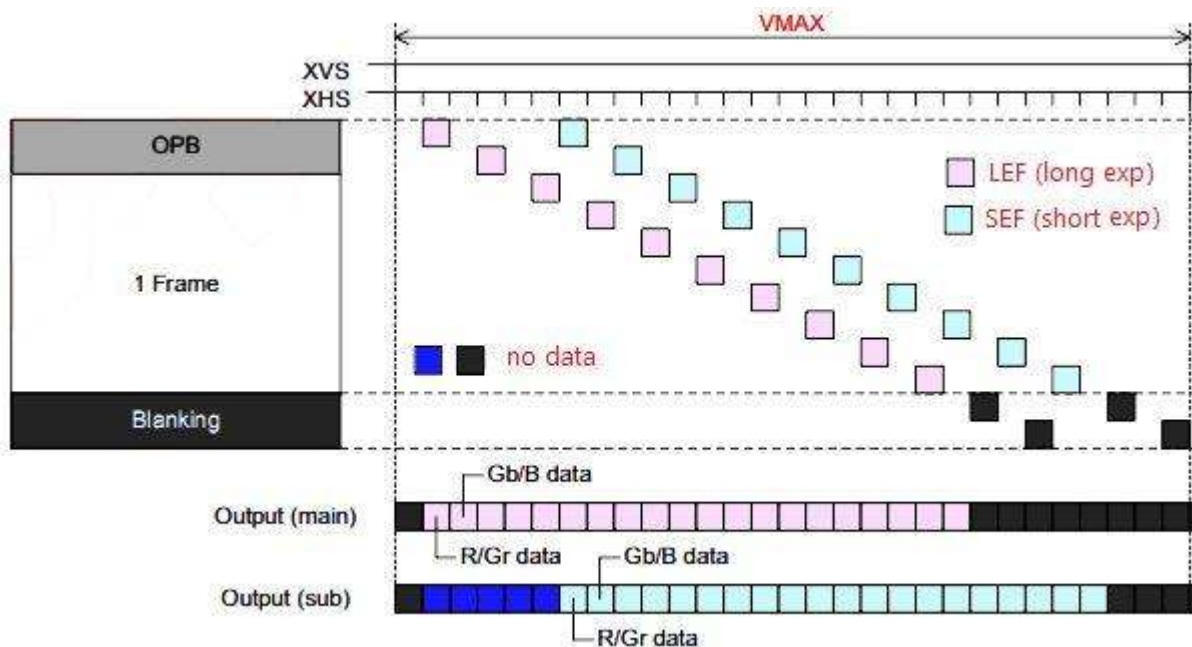
SONY支持WDR的方式稱為DOL(Digital OverLap)技術，最多支持三幀曝光。與Staggered WDR 技術不同點在於，它的輸出格式種使用了一些特別的標志數據，所以需要專用的邏輯電路進行解析。它支持兩種像素輸出方式，方式1是使用1個stream輸出，每行由長曝光+短曝光行交替，方式2是使用兩個stream並行輸出，如下圖所示。



知乎

首發於
無法抗拒的圖像技術

SONY DOL - single stream 交替輸出方式



SONY DOL - main/sub stream 並行輸出方式

仔細觀察SONY和OV的原理圖，很容易注意到長、短行是錯開一定間隔的，即最開始出來的數據全部是長行+dummy，最後出來的幾行全部是dummy+短行。

不難理解，ISP 做長、短曝光融合時，必須要同時取走相同行號、相同坐標位置的一對像素數據。因此，在sensor 端錯開輸出的行數據必須在進入ISP 之前**重新對齊**（re-align）。通常的做法是在ISP前端設計若干行line buffer 緩存長行數據，等短行數據也到達後，ISP從長行line buffer中取長曝光像素，同時從短行line buffer中取實時到達的短曝光像素。

第二個問題是緩存長行的line buffer設計成多大會比較合理。從SONY的原理圖上看，預留8行line buffer 似乎應該夠用了。如果未來出了一款sensor需要16行line buffer，則很不幸就不能支持在線模式了，只能先把長、短數據寫入內存，然後同時從內存讀取同一行號的長、短數據。

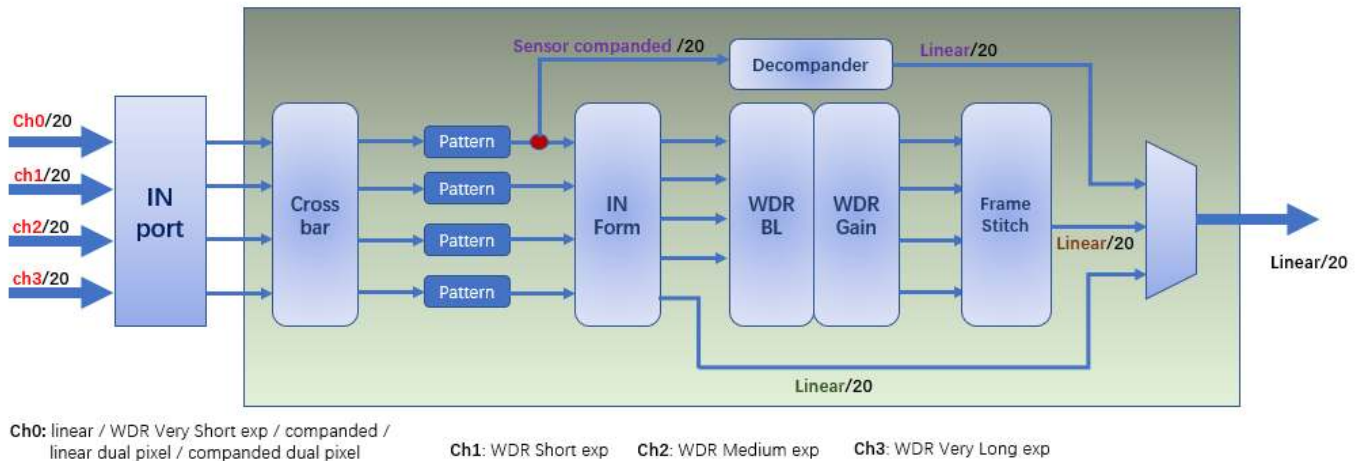


知乎

首發於
無法抗拒的圖像技術

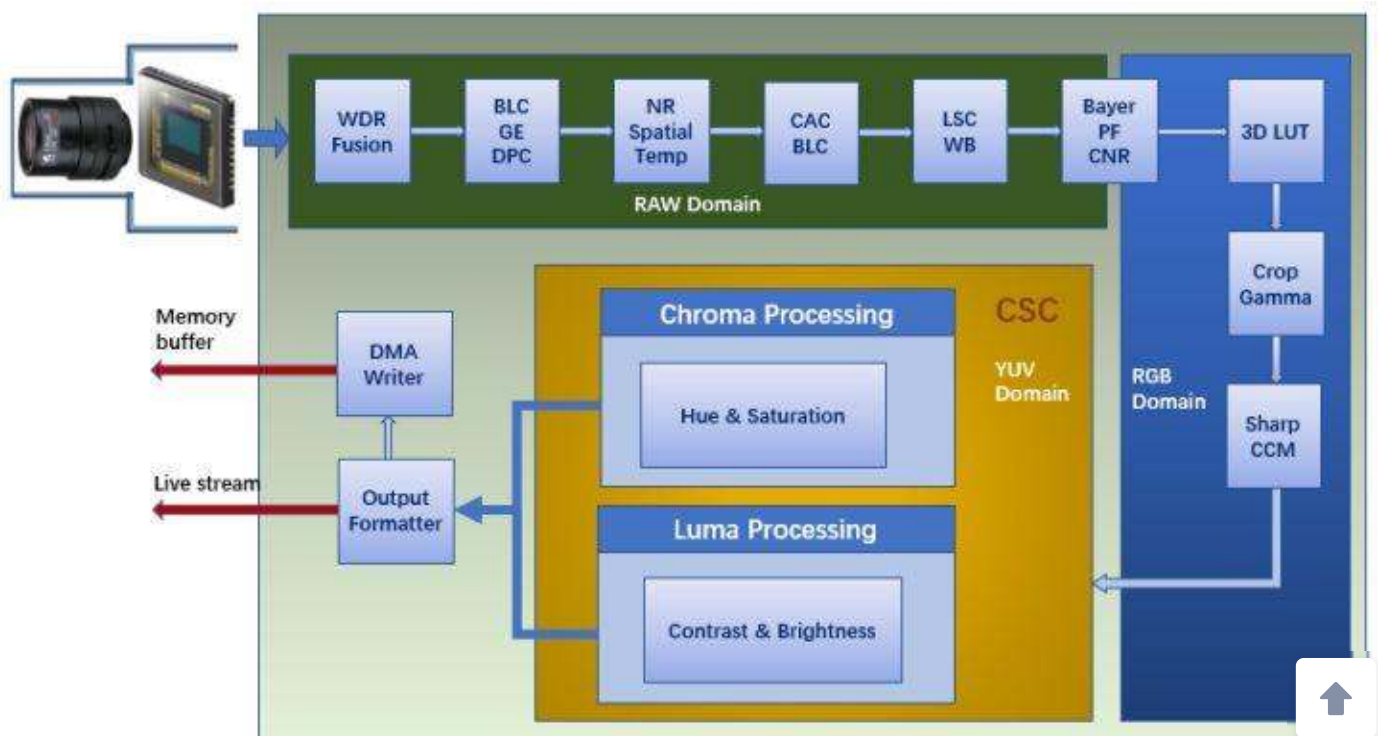
網絡算法時非常容易出現。一般ISP會允許5000個時鐘週期左右的等待時間，如果超過這個時間，ISP 就會產生時序錯誤，此時ISP 會發出一個硬件中斷，請求軟件干預。

下面是一個主流ISP方案對WDR輸入的處理流程圖。除了正常的線性輸入模式外，該ISP還支持最多四幀合成WDR，其中ch0支持線性輸入和WDR的最短幀，ch1/2/3支持WDR的短/中/長幀。



該ISP還支持companded和dual pixel數據格式，這是因為某些sensor輸出的不是線性數據，而是經過函數變換的數據，或者把兩個像素的值合並到一個16位數據中輸出，ISP也可以處理這些比較特殊的輸入格式。

下面是該ISP的完整流程圖，除了WDR功能之外的亮點是增加了空域和時域降噪模塊，增加了顏色處理（PurpleFringing/PF，ColorNoiseReduction/CNR）和3D顏色變換表（LUT），可以實現更好的色彩還原效果，以及應對高挑戰的低光夜視場景。



4K分辨率的圖像有800W個像素，RAW格式佔1600W字節，YUV422格式佔1200W字節。如果客官對1200W沒啥概念，可以想想在上海陸家嘴買一套135平大三房需要多少錢，差不多就是這麼大的壓力。其實還沒完，想想8K分辨率，也就是7680x4320，RAW格式佔6600W字節，這個壓力就大得不敢想像了，估計湯臣一品大平層也差不多夠了。

好了，想必客官已經清楚在內存中將這麼大量的數據搬來搬去不是一件很容易的事情了。為了減輕傳輸帶寬和存儲的壓力，支持4K 以上的芯片都會在DMA上設計一個壓縮算法。當DMA向內存中寫入數據時，實際進入內存的是壓縮後的數據。當DMA從內存中讀取數據時，用戶得到的是解壓縮後的數據。

Arm 出售的圖像壓縮技術叫AFBC，即Arm Frame Buffer Compression，這是一種基於脈沖編碼調制(Pulse Code Modulation, PCM) 技術實現的無損壓縮技術，典型情況下可以實現50% 左右的壓縮率，可以節省存儲空間和傳輸帶寬。有情報顯示，華為在其手機芯片中實現了基於小波變換的有損壓縮技術，壓縮效率應該更高。

在工程實踐種還存在另外一種做法，也是基於PCM技術，但壓縮後的數據不是緊密排列在一起，而是以塊為單位，壓縮數據仍放在自己所在的塊內，於是兩個塊之間會存在一些空洞。這樣做的好處是可以方便地找到每一個塊的數據，弊端是不能夠節省存儲空間，但仍然可以起到節省帶寬的效果。

色調映射 Tone Mapping

攝像機拍攝室外場景時，晴朗夏天的光照度可以達到10萬~20萬lux，理論上拍攝這種場景需要提供高達5000：1的動態範圍，在攝像機內部則需要使用至少13位的數據才能表示5000：1的動態範圍，在通用 CPU架構中使用16位整數則更加方便。由於數據在處理環節經常涉及除法、開方、指數等浮點運算，所以還需要預留若干個小數位以保持浮點精度，4位二進制小數可以提供0.0625精度，8位二進制小數可以提供0.0039精度。上述的主流ISP方案中使用20位數據。

當圖像在顯示設備上輸出時，普通的LDR顯示器只能提供256級灰度，按數量級是100:1的動態範圍。符合HDR10標準的顯示器可以提供1000:1的動態範圍，已經可以較好地還原自然場景的動態。如果攝像機的適配輸出設備是LDR顯示器，則攝像機的ISP內部需要完成從5000:1到100:1的動態範圍壓縮。

當WDR模塊完成多幀合成 (frame stitch) 後，接下來就需要對數據位寬進行壓縮以節約後續步驟的計算資源。比較合理的做法是採取逐級壓縮策略，比如在WDR模塊先壓縮到12位精度，經過CCM、Gamma 等顏色處理後進一步壓縮到10位精度，經過CSC模塊後進行最後一次壓縮得到最終的8位精度輸出。

從16/20位精度壓縮到12位精度的過程稱為色調映射，這一步驟的主要任務是壓縮圖像的動態範圍，將HDR圖像映射到LDR圖像，並盡量保證圖像細節不損失。



知乎

首發於
無法抗拒的圖像技術

色調映射的方法大致分為兩類，即全局算法和局部算法。

全局算法 (Global Tone Mapping , GTM)

全局算法可以理解為每幅圖像有一個顏色映射表，GTM算法通過查表的方法把一個輸入顏色映射為一個輸出顏色。有些算法對所有圖像都使用固定的表，有些算法則是針對每一幀圖像創建不同的表。

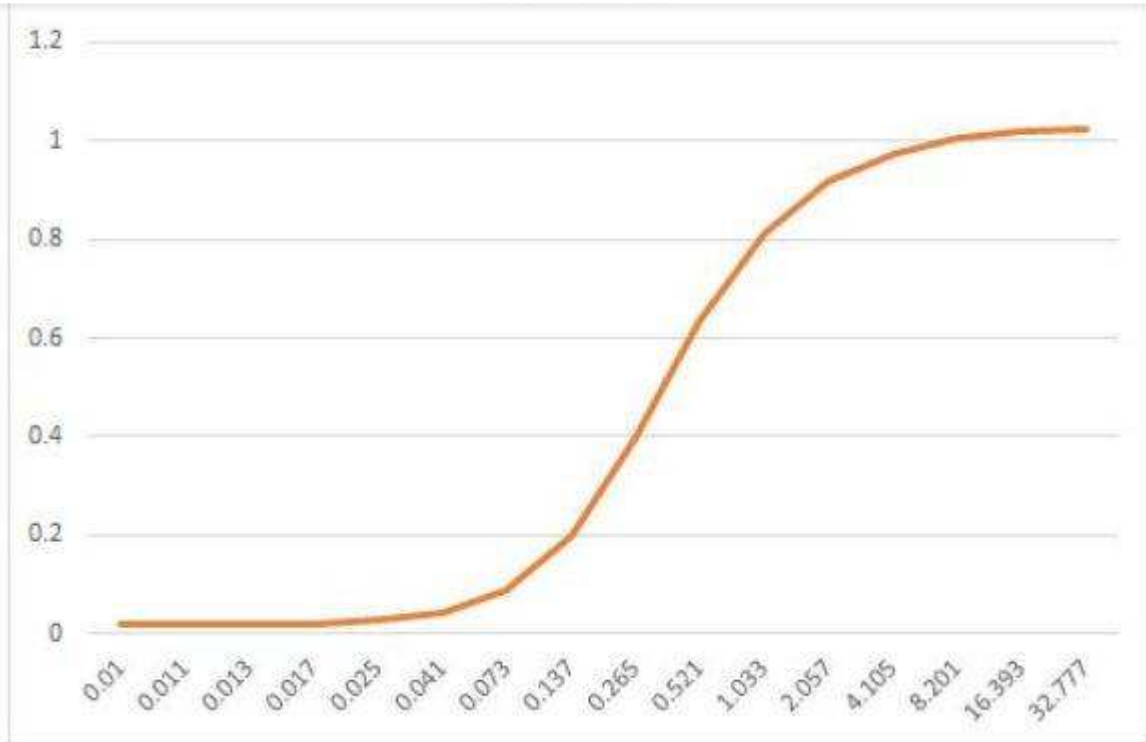
GTM算法特點：

1. 任意相同顏色的像素點，在映射後，還是相同的顏色；
2. 全局算法一般較簡單，速度快；
3. 全局算法的性能一般劣於局部方法；

存在算法：直方圖均衡化、Gamma、對數校正、直方圖規定化、分段灰度變換

一種基於Gamma的算法叫做Academy Color Encoding System (ACES)，它由美國電影藝術與科學學會提出，有人認為這是目前最好的一種GTM算法。ACES 本質上是一個通用的數據交換格式，既可以把不同的輸入設備轉成ACES，也可以把ACES在不同的顯示設備上正確地顯示。不管是LDR還是HDR都可以在ACES裡表達出來。ACES的轉換曲線如下圖所示。





代碼也很簡潔。

```

1 float3 ACES ToneMapping(float3 color, float adapted_lum)
2 {
3     const float A = 2.51f;
4     const float B = 0.03f;
5     const float C = 2.43f;
6     const float D = 0.59f;
7     const float E = 0.14f;
8
9     color *= adapted_lum;
10    return (color * (A * color + B)) / (color * (C * color + D) + E);
11 }

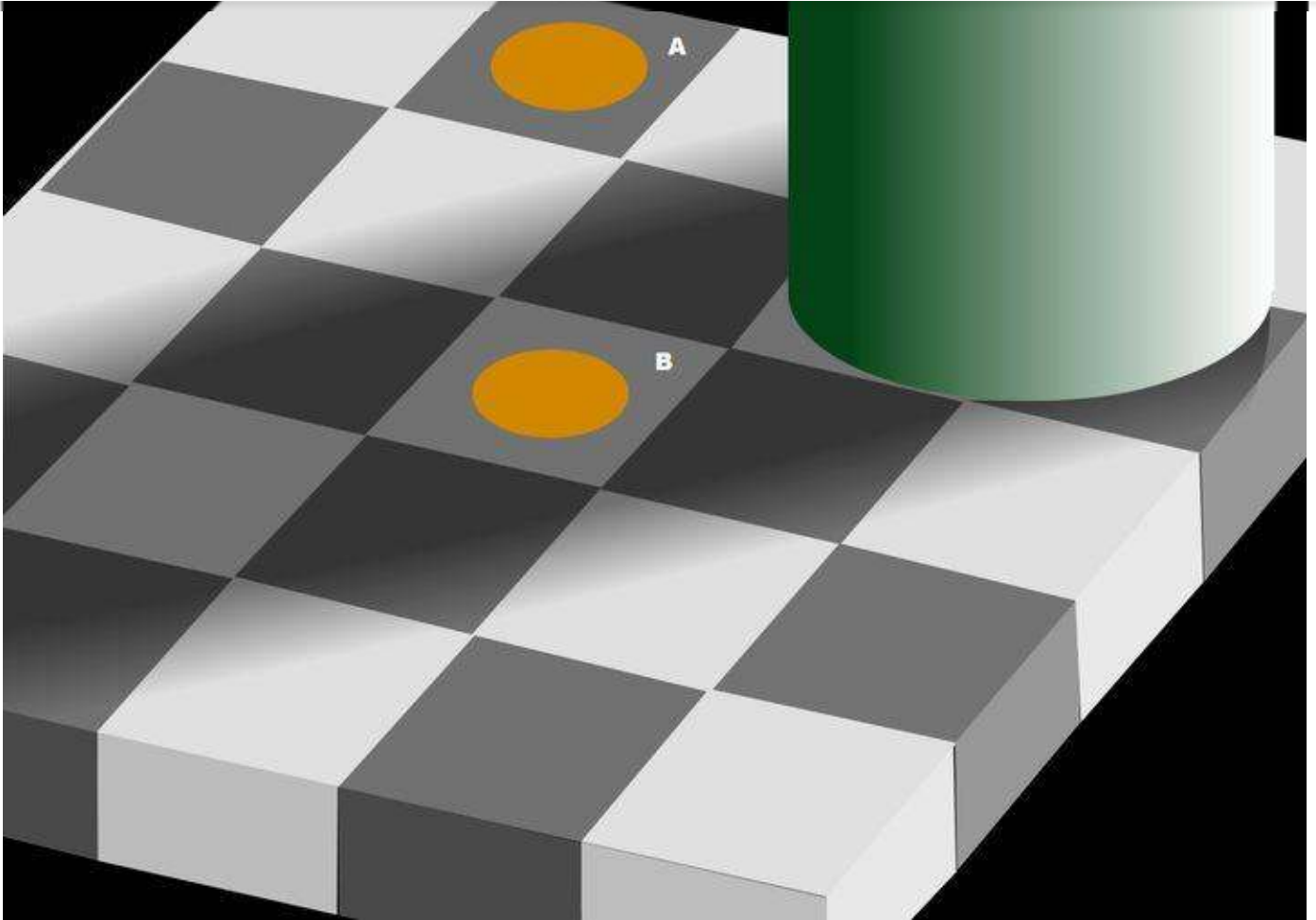
```

局部算法 (Local Tone Mapping , LTM)

人眼的感知特性具有局部性特點，舉例來說，在普通人眼看來下圖中的A色塊的亮度明顯低於B色塊的亮度。而事實真相是，這兩個色塊的真實像素亮度是一模一樣的，人眼感覺B更亮，主要是因為B的周圍是暗色塊，而A的周圍是亮色塊，人的知覺系統針對這種場景自動做了對比度提升。



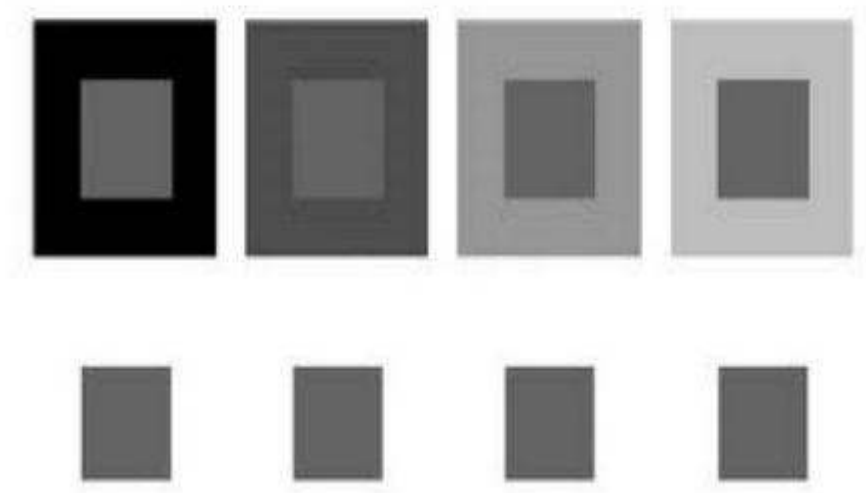
知乎

首發於
無法抗拒的圖像技術

這種現象在總體上可以歸結為人的知覺恆常特性，具體地說就是亮度恆常特性。下圖是另外一個亮度恆常的例子。在這個例子中，所有小灰色塊的真實亮度都是一樣的，但是如果周圍環繞的顏色不同，人對灰色塊的顏色知覺也不相同。



知乎

首發於
無法抗拒的圖像技術

局部算法借鑑了人眼的知覺原理，在映射一個像素時，不僅考慮該像素的絕對值，還會考慮該像素周圍區域的平均亮度值，將對比度大的像素映射為高亮，對比度小的像素映射為低亮，往往可以取得更好的效果。

LTM算法特點：

1. 映射前顏色相同的像素點，映射後顏色可能不同
2. 局部算法一般較全局方法更復雜，速度相對較慢；
3. 局部算法的性能一般優於全局方法；
4. 會出現光暈等現象

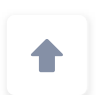
存在算法：分塊中值直方圖，基於Retinex原理的算法等

關於色調映射的更多內容可參考主題文章

劉斯寧：Understanding ISP Pipeline -
Tone Mapping
zhuanlan.zhihu.com



RAW域處理

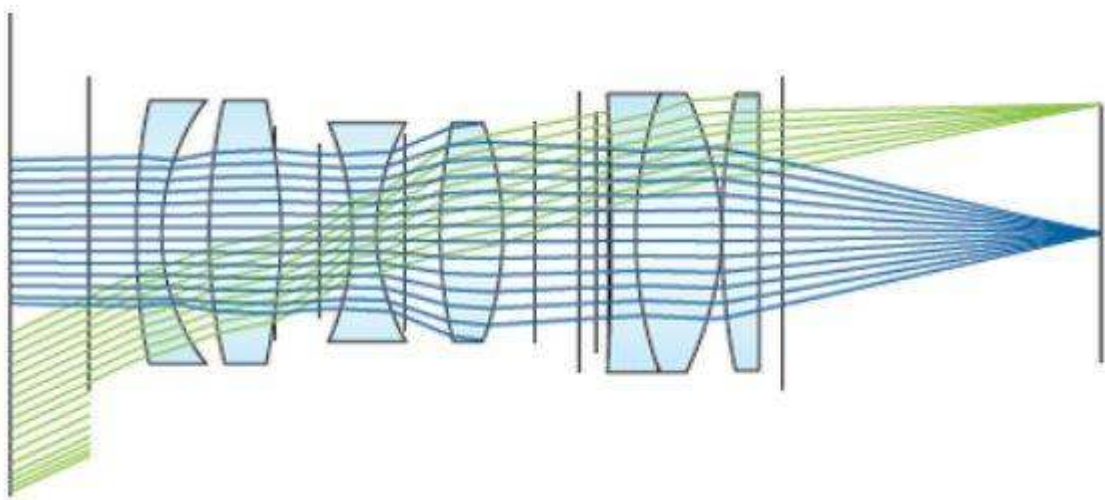


鏡頭陰影有兩種表現形式，分別是

- Luma shading，又稱vignetting，指由於鏡頭通光量從中心向邊緣逐漸衰減導致畫面邊緣亮度變暗的現象。
- Chroma shading，指由於鏡頭對不同波長的光線折射率不同引起焦平面位置分離導致圖像出現偽彩的現象。

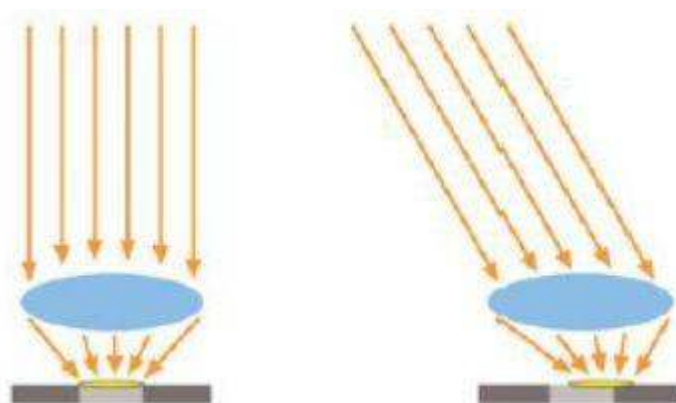
Vignetting原理

1. 畫面邊緣鏡頭能量衰減



如上圖所示，由於鏡頭中都會存在多處光闌，當入射光線偏離光軸角度較大時，部分光線就會被光闌遮擋而不能參與成像，因此越靠近sensor邊緣的像素接收到的曝光量就越低。

2. 邊緣像素微透鏡和感光面的錯位



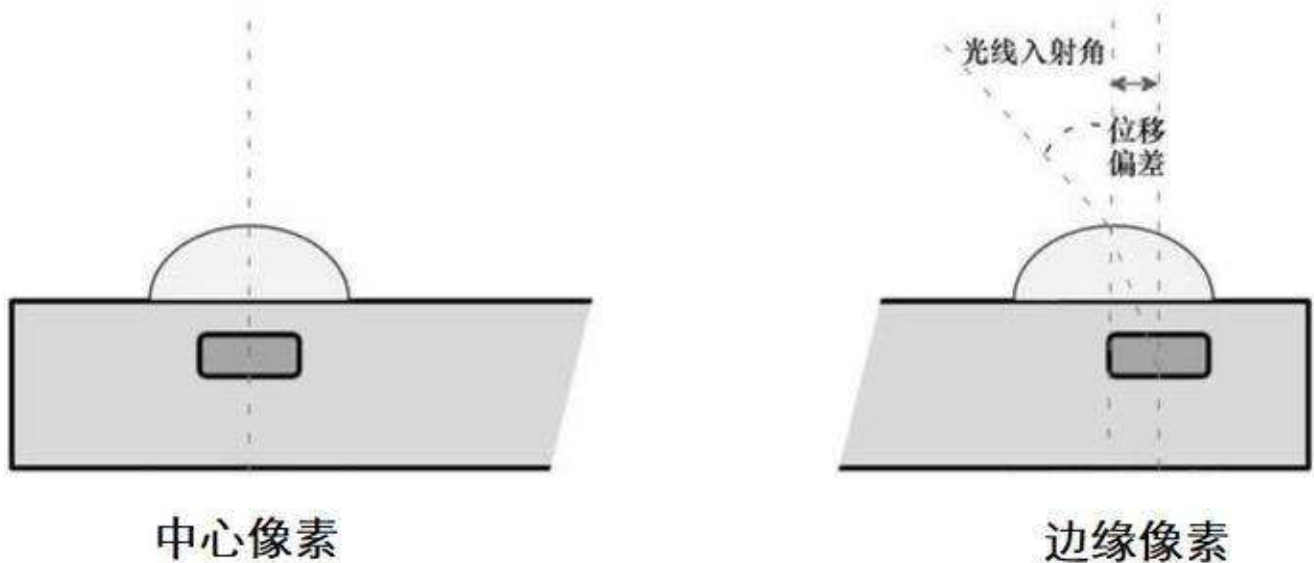
焦點錯位導致能量損失



知乎

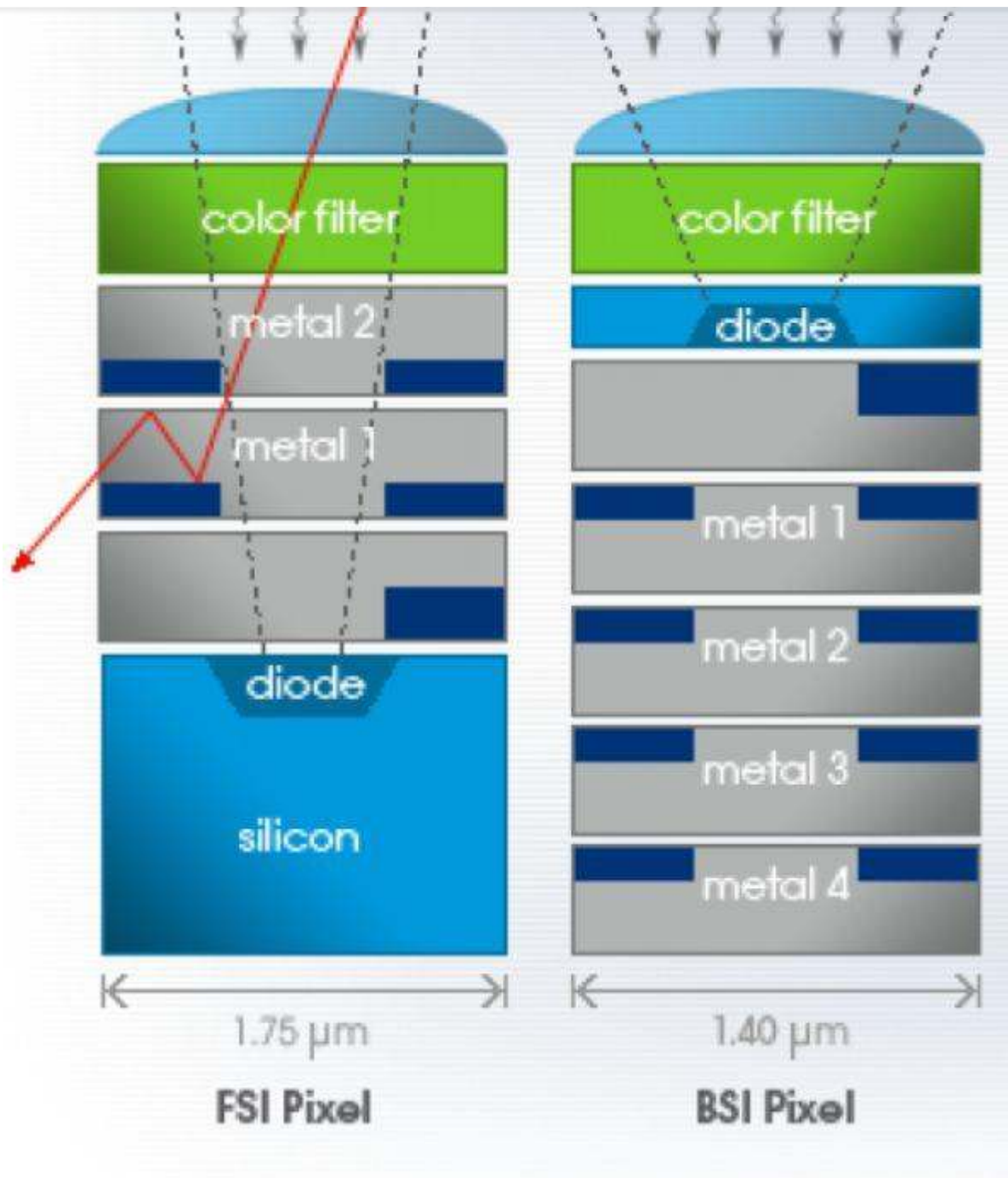
首發於
無法抗拒的圖像技術

感光面的面積一點點，這樣越往邊緣微透鏡與感光面之間的錯位就越大，剛好可以補償入射光線角度增大導致的焦點偏移，使光線可以更好地聚焦到感光面上，如下圖所示。



不過深入研究會發現，這個補償辦法其實也是有侷限的，如果sensor採用的是下圖左所示的FSI工藝（前照式），從像素微觀結構來看，當入射光線角度比較大時，會有較多光線與像素中的金屬布線層發生吸收、散射從而產生損失，單純移動微透鏡的位置並不能有效解決這個問題。但是，如果像素採用的是右圖所示的BSI工藝（背照式），因為布線層在硅片的另外一側，所以光線損失會少，補償效果更加有效。

知乎

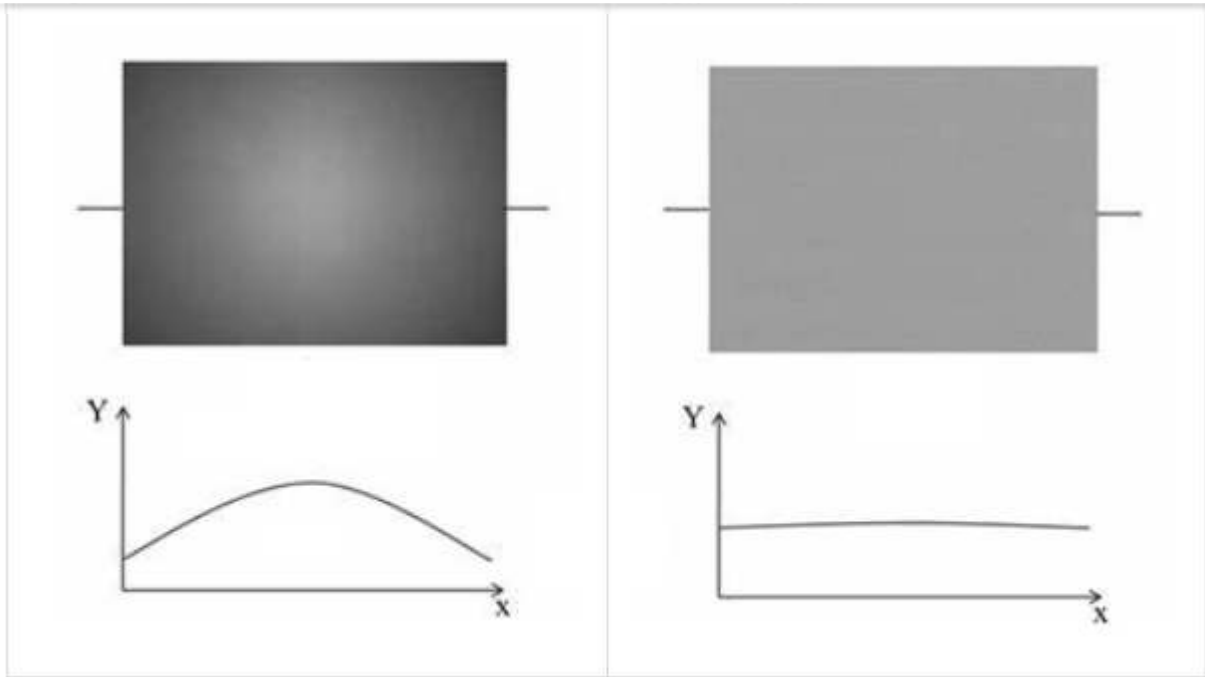
首發於
無法抗拒的圖像技術

Vignetting是由鏡頭引起的現象，所以LSC校正也是針對特定鏡頭的。若果產品的適配鏡頭發生變化，原則上需要重新進行LSC校正。

另外，Vignetting現象在sensor靶面較大、鏡頭焦距較短時表現更加明顯。採用非球面鏡頭通常可以改善vignetting。

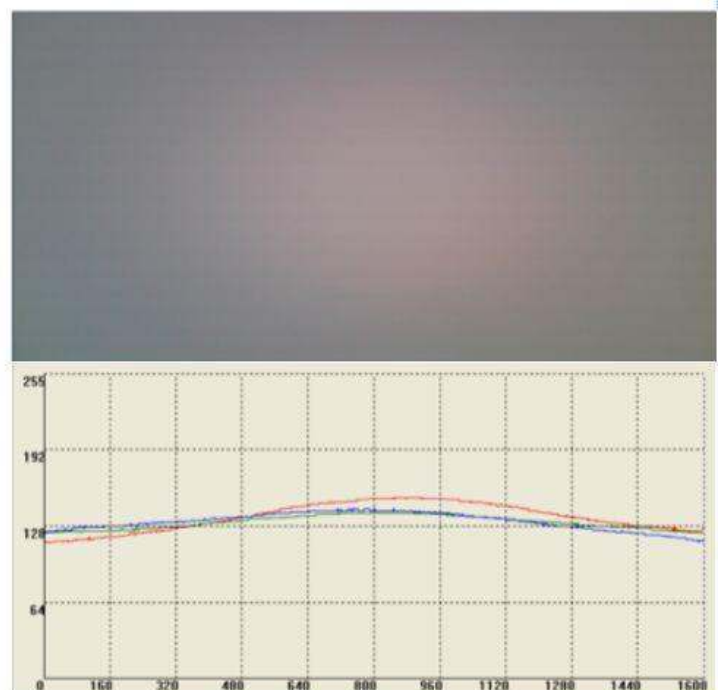
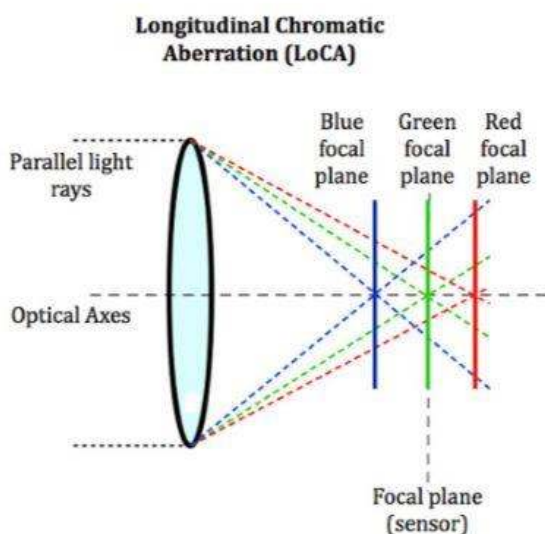


知乎

首發於
無法抗拒的圖像技術

Chroma shading 原理

鏡頭對不同波長的光線折射率不同會導致色差問題，即不同波長的焦點在空間上不重合，導致焦平面分裂為三個不完全重合的曲面，這會破壞圖像的白平衡，使圖像出現偽彩，如下圖所示。根據 sensor 所處的前後位置不同，偽彩可能偏紅也可能偏藍。



Chroma shading 一般主要通過鏡頭選型來控制其影響。如果ISP支持chroma shading校正可以通過標定三個顏色平面的增益來修正。為了控制標定表格的存儲空間，通常只標定MxN



空域濾波器 (Spatial Filter)

對圖像降噪的主要方法是使用空域濾波器對圖像進行濾波。濾波操作通常是針對以某個像素為中心的濾波窗口上進行的，濾波窗口的大小與具體的算法有關，常用的大小有3x3、5x5、7x7等尺寸。濾波操作在數學上稱為卷積，需要使用一個與濾波窗口大小一致的卷積核，卷積核的每個元素代表一個權重，與對應位置的圖像像素值相乘，然後所有乘積累加到一起就是濾波後的結果。

卷積濾波用公式表示是，

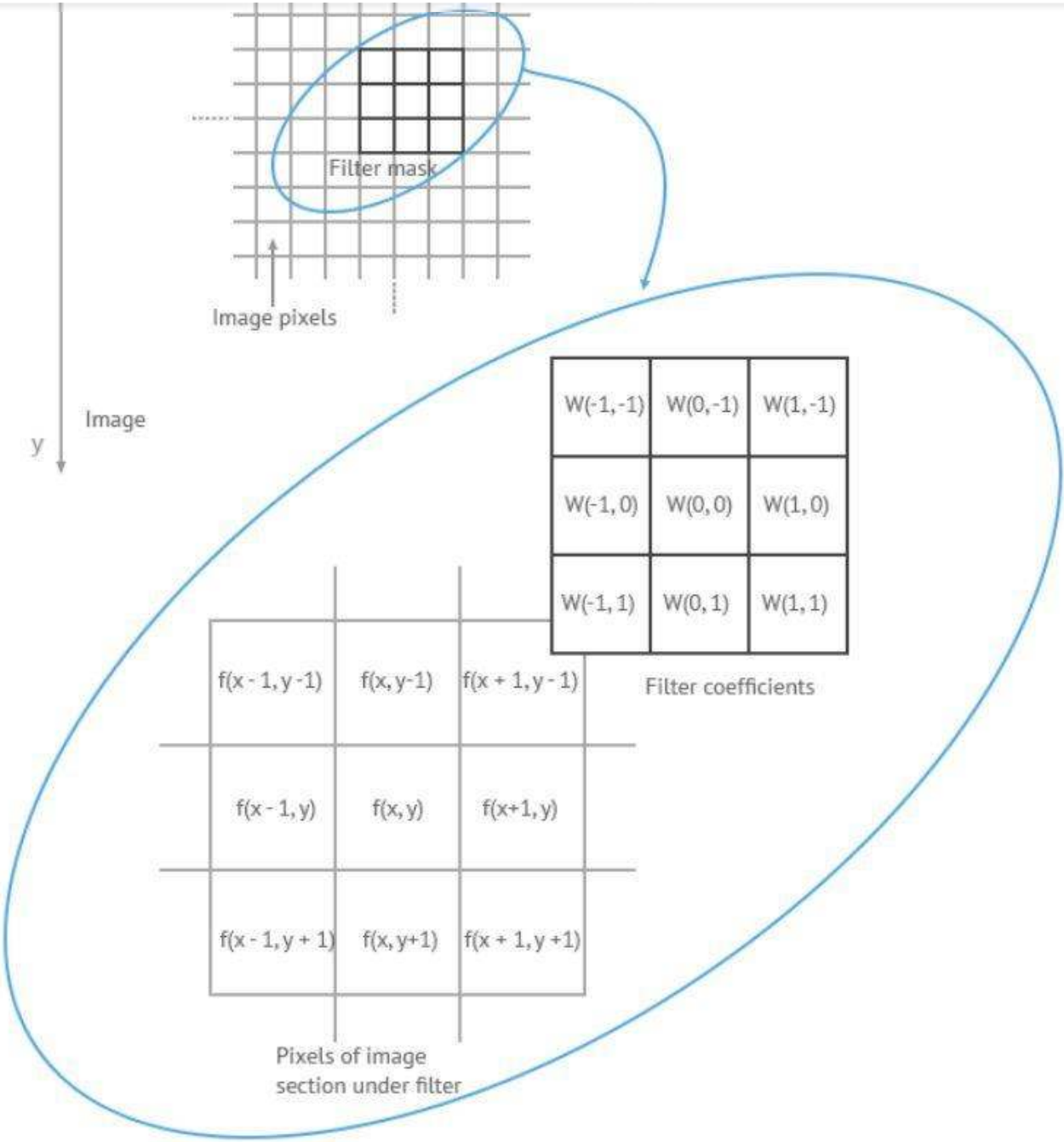
$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$

下面是卷積濾波操作的示意圖。



知乎

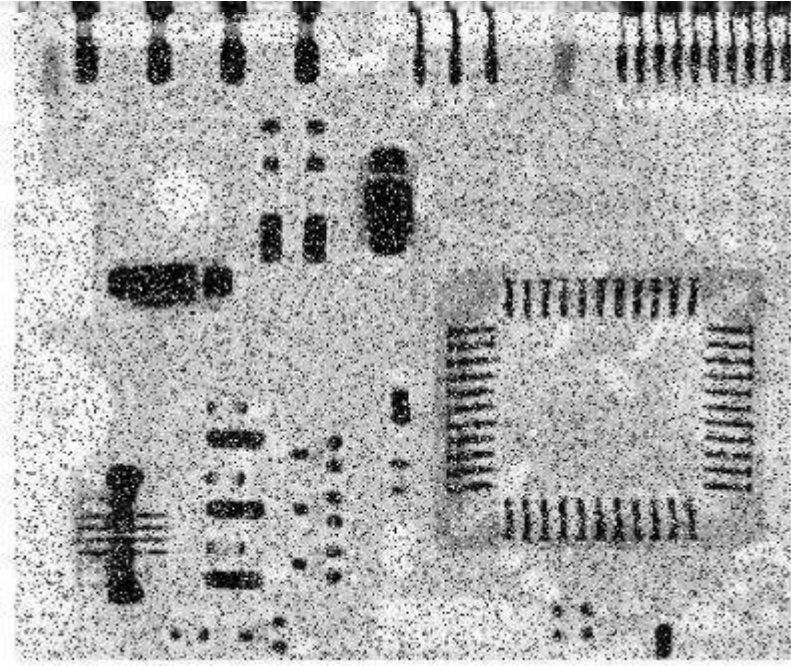
首發於
無法抗拒的圖像技術



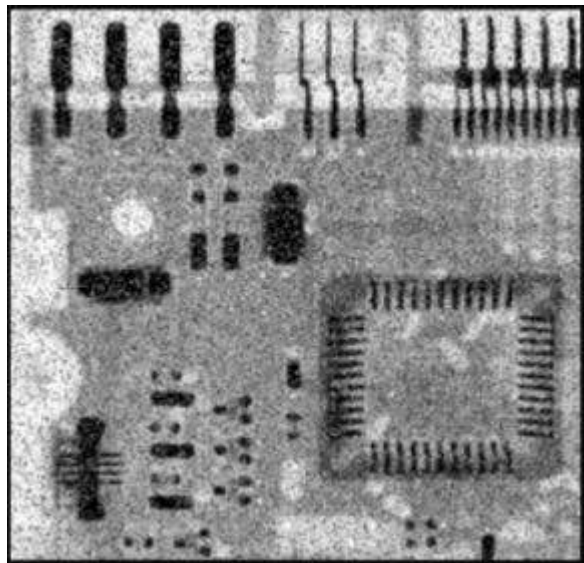
原始圖像：



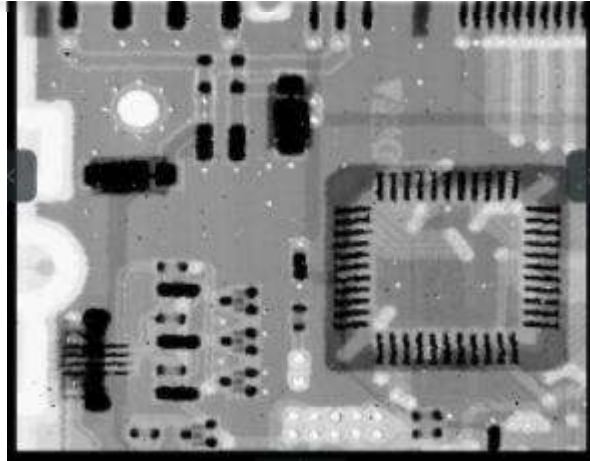
知乎

首發於
無法抗拒的圖像技術

均值濾波結果：



中值濾波結果：



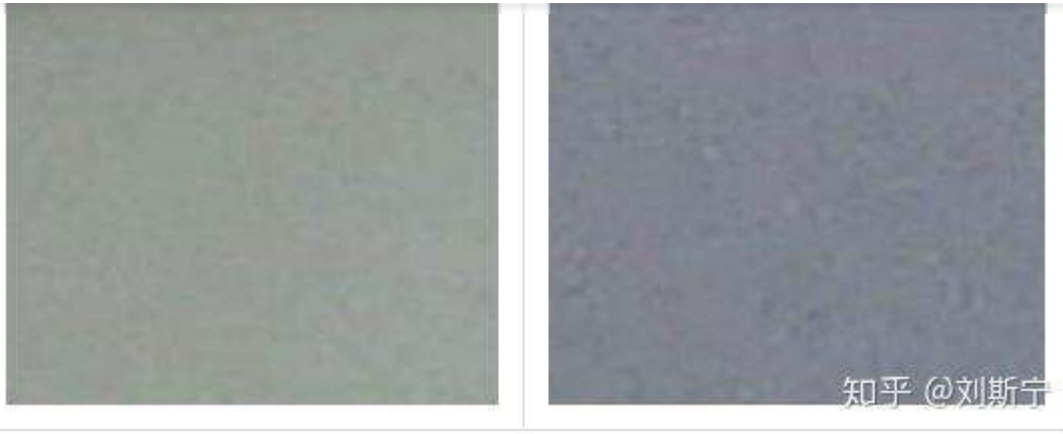
RAW域降噪

Sensor輸出的RAW圖像本身是攜帶了噪聲的，前面提到過sensor噪聲的種類主要包括熱噪聲、光散粒噪聲、讀出噪聲、固定模式噪聲等。當sensor溫度較高、增益較大、環境較暗的情況下各種噪聲會變得更加明顯，成為影響圖像質量的主要因素。在下圖所示的例子中，左圖光線較充足，圖像清晰無噪點，而右圖中光照不足，camera 需要提高增益才能使畫面達到正常亮度，同時也放大了噪聲，圖像出現明顯的噪點。



除了Sensor圖像本身攜帶的噪聲之外，圖像每次會經過ISP模塊的處理之後都會引入一些新的噪聲，或者對原有噪聲進行了放大。以LSC模塊為例，LSC校正的實質是在輸入圖像上乘以一個與像素位置有關的增益系數以補償光信號的衰減，而補償的規律是越遠離圖像中心的地方增益越大。根據噪聲傳播的基本原理，當增益系數大於1時，圖像中的噪聲是與信號一起被同步被放大的。另外，由於ISP所用乘法器的精度是有限的，每做一次乘法就會重新引入一次截斷誤差，這是新增的噪聲來源，所以經LSC處理後圖像的整體噪聲水平會有所增加，而且在圖像的邊緣處表現會更加明顯，典型的效果如下圖所示。





Shading固然是不好的，需要校正，但是為了校正shading而給圖像引入噪聲同樣也不好的，所以人們需要權衡在多大程度上校正shading能夠收到滿意的效果。這是在主觀圖像質量調試階段需要考慮的問題之一。

研究發現，噪聲在ISP流水線各模塊中會不斷產生、傳播、放大、改變統計特性，對圖像質量的影響會越來越大，而且越來越不容易控制。因此處理噪聲的基本原則是越早越好，隨時產生隨時處理，盡可能將問題消滅在萌芽狀態。目前主流的ISP產品中一般會選擇在RAW域、RGB域、YUV域等多個環節設置降噪模塊以控制不同類型和特性的噪聲。在YUV域降噪的方法已經得到了廣泛的研究並且出現了很多非常有效的算法，但是在RAW域進行降噪則因為RAW數據本身的一些特點而受到不少限制。主要的限制是RAW圖像中相鄰的像素點分別隸屬於不同的顏色通道，所以相鄰像素之間的相關性較弱，不具備傳統意義上的像素平滑性，所以很多基於灰度圖像的降噪算法都不能直接使用。又因為RAW數據每個像素點只含有一個顏色通道的信息，所以很多針對彩色圖像的降噪算法也不適用。

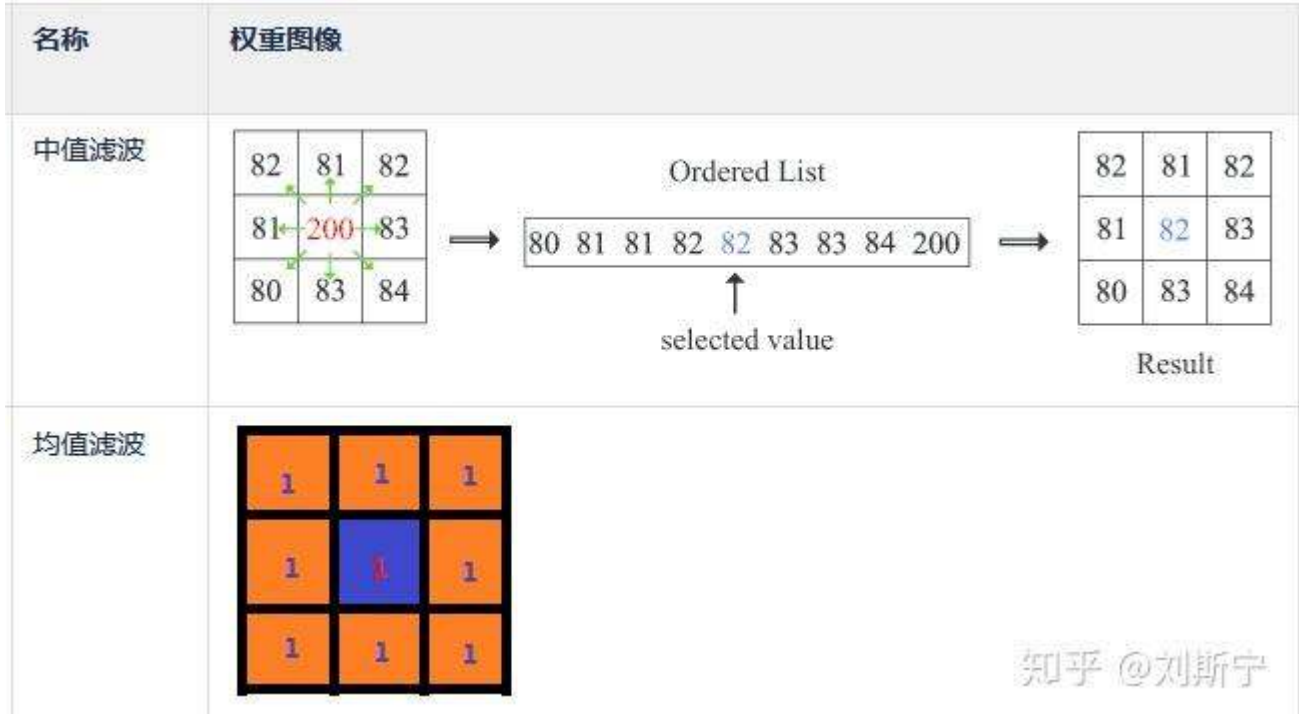
和很多圖像處理算法一樣，降噪即可以在空域(spatial domain)上實現，也可以在頻域(frequency domain)上實現，比較有常用頻域方法有傅裡葉變換，離散餘弦變換(DCT)，小波變換，多尺度幾何分析等。隨著人工智能技術的發展，近些年來還湧現了一批基於深度學習技術實現的降噪算法。後面提到的這些方法雖然都有不錯的性能，但是對算力要求都比較高，並不一定適合處理高分辨率的實時視頻流，所以在ISP產品中應用的並不廣泛，目前適合ISP應用的降噪算法還是以經典低通濾波器的改進版本更為常見。

目前在RAW域降噪基本都需要將RAW圖像按照顏色分成四個通道(R,Gr,Gb,B)，然後在各個通道上分別應用濾波器進行平滑，根據濾波器的特點和複雜度大致可以分成以下幾類：

1. 經典低通濾波器，如均值濾波、中值濾波、高斯濾波、維納濾波等。這類方法的優點是比較簡單，佔用資源少，速度快，缺點是濾波器是各向同性的，容易破壞圖像中的邊緣。另外由於沒有考慮顏色通道之間的相關性所以也容易引入偽彩等噪聲，而人眼對這種顏色噪聲是比較敏感的。
2. 改進的經典濾波器，如Eplison濾波、雙邊濾波(bilateral filter)，在經典濾波器的基礎上增

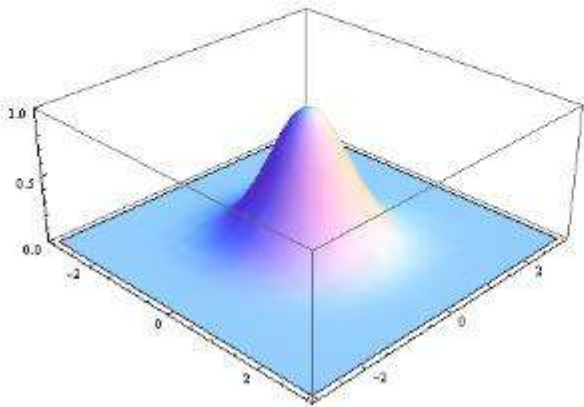
與經典濾波器基本相同。

3. 引導濾波器(guided filter)，由何凱明博士早期提出的一種算法，引入了引導圖像的概念。對於任一顏色通道的圖像，以當前位置像素 $P(x,y)$ 為中心，在一個固定大小的濾波窗口內為 $P(x,y)$ 的所有鄰近像素 $\{P(i,j)\}$ 計算權重 $\{W(i,j)\}$ ，計算權重的方法是以某個引導圖像 I 作為參考，在引導圖像的對應濾波窗口內，凡是與像素 $I(x,y)$ 性質“類似”的像素都得到較大的權重，與 $I(x,y)$ 性質“相反”的像素則得到較小的權重。舉例來說，如果當前像素 $P(x,y)$ 是亮的，則濾波窗口 $\{P(i,j)\}$ 中的全部亮像素會參與平滑，平滑結果仍是亮的，同理，如果與 $P(x,y)$ 相鄰的某個像素 $P(x',y')$ 是暗的，則對應的 $\{P(i',j')\}$ 中的全部暗像素會參與平滑，平滑結果仍是暗的，這樣 $P(x,y)$ 與 $P(x',y')$ 之間的明暗邊界就得到了保持。這種方法的基本假設是圖像各通道的顏色梯度分佈與引導圖像是一致的，如果假設不成立，則從引導圖像計算出的權重反而容易破壞其它通道中的邊緣。有人建議使用demosaic還原出的G通道圖像作為引導圖像會得到較理想的效果，因為G通道集中了圖像的大部分能量，也包含了豐富的梯度信息，是比較理想的候選對象。
4. 基於塊匹配的濾波算法，利用圖像的自相似特性，在以當前像素為中心的一個濾波窗口內找到與當前塊最相似的幾個塊，當前像素的濾波值即等於幾個相似塊的中心像素的加權平均值。此類算法以非局部均值濾波(Non-Local Means)和BM3D(Block Matching 3D)算法為代表，它們的優點是平滑性能和邊緣保持性能很好，缺點是計算量很大，資源消耗大，不太適合處理實時視頻。已有情報顯示，華為在某些型號手機芯片中實現了BM3D算法用於對照相圖片做降噪處理，估計這主要是為了體現行業龍頭企業佔領技術高地的決心。



知乎

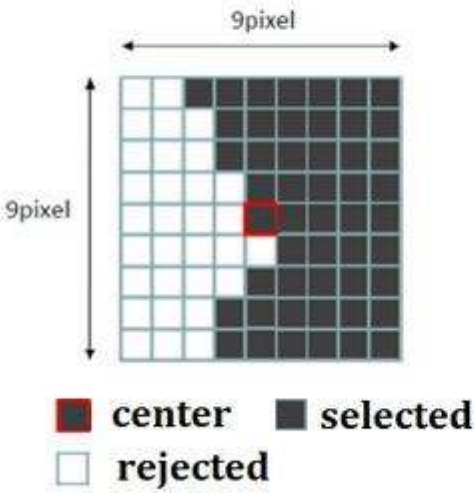
首發於
無法抗拒的圖像技術



知乎 @刘斯宁

Epsilon滤波

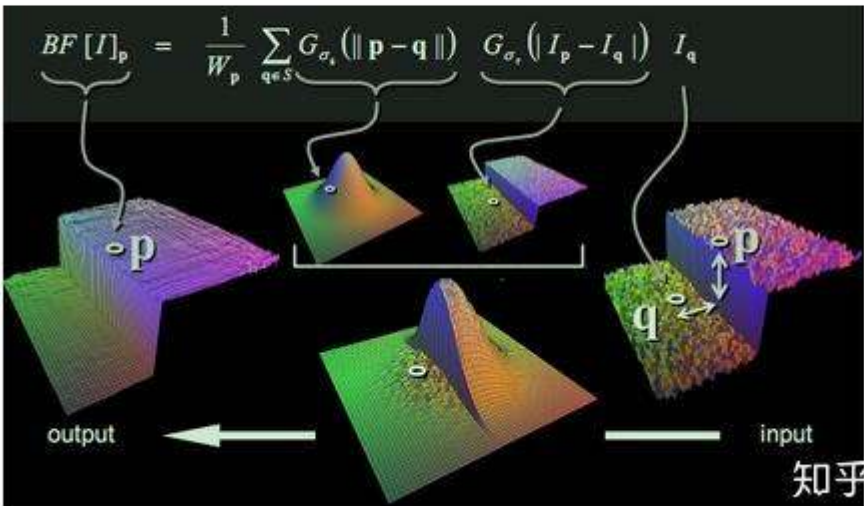
$$w(i,j,x,y) = |p(i,j)-p(x,y)| < \text{threshold} ? 1:0$$



知乎 @刘斯宁

知乎

首發於
無法抗拒的圖像技術



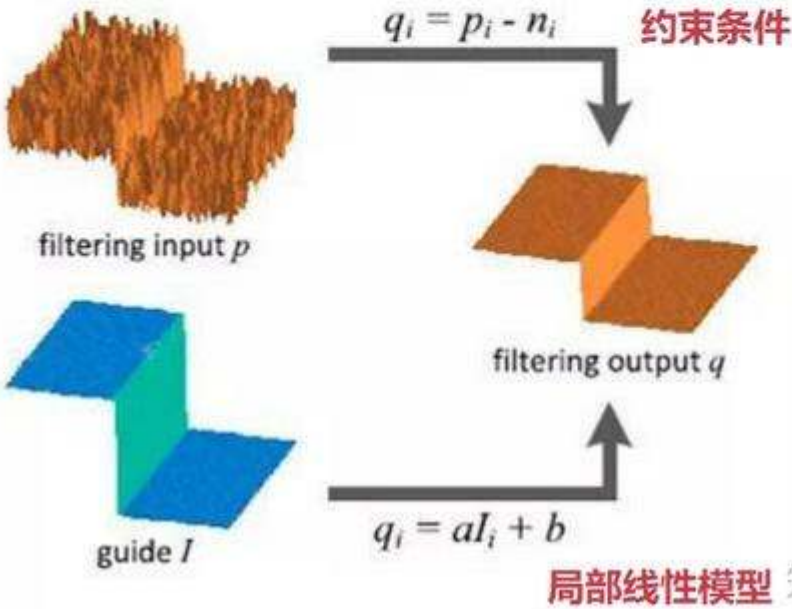
知乎 @刘斯宁

引导滤波

$$q_i = a_k I_i + b_k, \forall i \in \omega_k.$$

$$a_k = \frac{\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \epsilon}$$

$$b_k = \bar{p}_k - a_k \mu_k.$$



知乎 @刘斯宁

$$\|V(x) - V(y)\|^2 = \frac{1}{d^2} \sum_{\|z\|_\infty \leq ds} \|v(x+z) - v(y+z)\|^2$$
$$Z(x) = \sum_y \exp(-\frac{\|V(x) - V(y)\|^2}{h^2})$$

知乎 @刘斯宁
Go to

BM3D

<https://www.cnblogs.com/whw19818/p/5765990.html>

知乎 @刘斯宁

下圖是雙邊濾波的效果示例，可以看到濾波和邊緣保持的效果都是很理想的。

知乎

首發於
無法抗拒的圖像技術

關於降噪算法的更多討論可參考本專欄的另一篇主題文章。

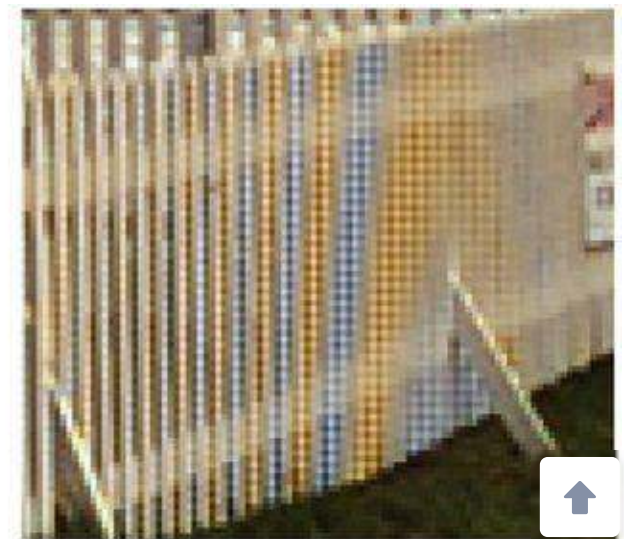
劉斯寧：Understanding ISP Pipeline -
Noise Reduction

zhuanlan.zhihu.com



Bayer Demosaic

RAW域的最後一步處理是Demosaic，將像素從RAW域變換到RGB域進行下一階段的處理。Demosaic 算法的主要難點在於，RAW域的任何一個像點（photosite）只包含一個真實的採樣值，而構成像素（R,G,B）的其它兩個值需要從周圍像點中預測得到。既然是預測，就一定會發生預測不准的情況，這是不可避免的，而預測不准會帶來多種負面影響，包括拉鏈效應（zipper artifacts），邊緣模糊，顏色誤差等。

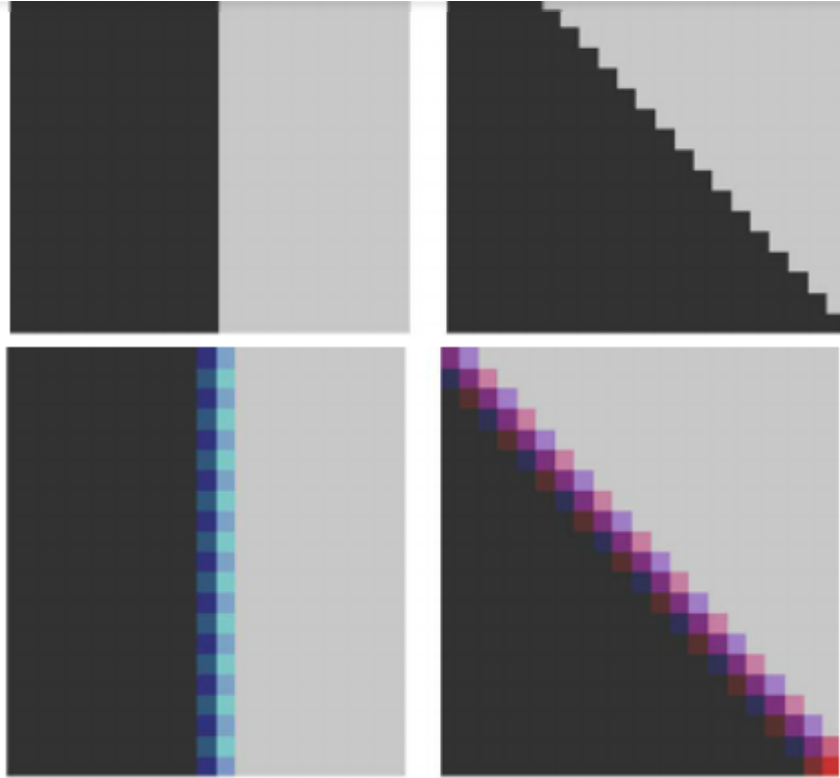




Demosaic 偽彩



知乎

首發於
無法抗拒的圖像技術

Demosaic 拉鏈效應，邊緣模糊，偽彩

所以Demosaic 算法的主要挑戰就是盡量提高算法的準確性，減少圖像邊緣損失和顏色誤差。

下面是引用一位知友的文章關於Demosaic的算法簡介。

凹凸思：Image Process PipeLine 之
DM (下)

zhuanlan.zhihu.com



ISP Tuning

to be continued...

劉斯寧：Understanding Image Quality
Tuning

zhuanlan.zhihu.com



圖像信號處理器ISP (Image Signal Processor)

Camera

▲ 贊同 196 ▼

💬 55 條評論

➦ 分享

♥ 喜歡

★ 收藏

📄 申請轉載

...

文章被以下專欄收錄



無法抗拒的圖像技術

推薦閱讀

Understanding ISP Pipeline
- Demosaicking

簡介人眼種存在三種感光錐細胞，分別對紅、綠、藍三種波段敏感。因此描述一個人眼可見的顏色需要且僅需要三個份量，這三個份量能夠支撐起一個顏色空間，該空間包含了人眼能夠知覺的所有顏色...

劉斯寧 發表於無法抗拒的...

Understanding ISP Pipeline
- Sharpen

什麼是銳度銳度（acutance）常用於描述邊界處圖像信息過渡的快慢。高反差圖像過渡速度非常快，可以形成非常明確的邊緣，而低反差圖像存在一定的過渡緩沖，表現在成像上就是模糊的邊緣。由...

劉斯寧 發表於無法抗拒的...

Under
- Cont

對比度要集中白的像果圖像了豐富度。如劉斯寧

55 條評論

⇌ 切換為時間排序

寫下你的評論...



大碗练习生

2020-03-14

大半夜看到这么用心的文章，让本不富裕的睡眠，雪上加霜[捂脸]，想请问老师，3DNR在ISP内部是个怎样的flow呢，如果通过保存多帧数据写入DDR，再读回处理生成一帧数据，这样3DNR的功能对core时钟就很不友好了，相同的帧率，时钟几乎翻倍，还是说读回也有类似virtual ID的概念可以pipeline

👍 1



知乎

首發於
無法抗拒的圖像技術

窗口区域) 进行实时运算, 没有回头再处理的问题。复杂一点的方案是保存两帧, 当第三帧来的时候, 第二帧与前后两帧进行运算, 输出第二帧。

👍 赞



大碗练习生 回复 刘斯宁 (作者)

2020-03-28

感谢大佬细致的评论, 期待更多优质文章

👍 赞



spongespongeyang

03-26

老师您好 想请教一下 为什么说 “如果信号处理的没有要求在线性空间做 blacklevel留在图片上面对去噪有一些的好处”

👍 赞



贾凯

03-25

请将大神FPN是在哪个阶段?

👍 赞



Lele wei

03-06

ISP方面比较好的入门文章, 学习了谢谢

👍 赞



非法字符

02-24

请问acetonemapping曲线的横坐标是什么含义呢?

👍 赞



嗯一起加油呵

01-29

感谢分享! 请问ISP里RGB2YUV会转换成 full range还是limited range (tv range)的yuv呢? 谢谢!

👍 赞



刘斯宁 (作者) 回复 嗯一起加油呵

01-29

有些ISP是可选的, 似乎用tv range居多

👍 1



嗯一起加油呵 回复 刘斯宁 (作者)

01-30

多谢! 采集到的图像后续想做处理和渲染, 需要转回RGB, 因此想要确认下。

👍 赞



丁金亮

01-19

知乎

首發於
無法抗拒的圖像技術

HE)，gamma 是不是就白做了。。。。

 赞

刘斯宁 (作者) 回复 不含空格

01-20

海思的做法是先做pregamma，然后做tone mapping，最后做正式的gamma

 赞

不含空格 回复 不含空格

01-20

嗯嗯 感谢

 赞

伍哲

01-15

可以再深入一些

 赞

奋斗青年

2020-11-10

老师的文章写得太好了，能将下 HISI的3DNR降噪吗，看海思3DNR降噪文档，一点都看不懂！又是时域，空域，色域降噪,,,,,,好难

 赞

洪晖

2020-09-07

博主你好，请问yuv域的降噪是怎么做的呢？

 赞

刘斯宁 (作者) 回复 洪晖

2020-09-07

低通滤波，Y和UV分开滤波，复杂一点的会加入边缘检测算子，边缘两侧分开滤波，保证边缘不损失

 赞

洪晖 回复 刘斯宁 (作者)

2020-09-08

那为什么要做低通滤波呢？

 赞[查看全部 6 条回复](#)

李思奇

2020-08-13

写的真好，关键模块都涉及了[赞同]

 赞

知乎

首發於
無法抗拒的圖像技術

ISP内部有自己的CPU core和操作系统，跑驱动和firmware算法库，对外面的Host CPU又是作为一个外设？这样理解对吗

 赞

刘斯宁 (作者) 回复 webbsun898

2020-07-28

是的

 赞

栗子脆脆脆

2020-07-08

你好，我想请问一下，为什么sensor出来的RAW图像和经过ISP出来的JPEG图像尺寸不一样呢？即使在PS中直接打开RAW图也是和原来的RAW数据宽高不一样？

 赞

刘斯宁 (作者) 回复 栗子脆脆脆

2020-07-10

ISP一般都会支持crop/scaler，会对输入图像进行裁剪和缩放，以输出标准分辨率，或满足下游模块的对齐要求。比如JPEG要求图像宽度必须是8的倍数，H.264要求16的倍数，所以经常在ISP环节就预先处理了。

 1

环保园一哥

2020-06-23

优秀（赞）

 赞

晓白

2020-06-18

谢谢分享

 赞

AomanHao

2020-06-17

写得特别好，向您学习，respect

 赞

识图行者

2020-03-17

RGB图像降噪是不是要根据色别分做3次呢？还是转YUV再做降噪呢

 赞

刘斯宁 (作者) 回复 识图行者

2020-03-27

降噪不是单选题，YUV域降噪最成熟，RGB降噪也是当前的热点，挑战更大的是Bayer域降噪，效果最好，但是技术也更复杂，因为Bayer域颜色信息不完整，亮度不连续。一

实用的ISP会在多个环节插入降噪节点，所以这是一个多选题！



知乎用户

2020-02-17

ISP离线模式是不是有可以同时处理多路camera图像的好处（类似于编码器）？只是这个分时复用是怎么控制AE的？网上一直没有相关的资料。不知知友有没有碰到过。

赞



刘斯宁 (作者) 回复 知乎用户

2020-02-18

如何控制AE 和离线模式没有任何关系。AE 是软件逻辑的问题，在ISP看来数据是以帧为单位的，ISP并不关心这一帧数据是属于哪一路的，camera 的控制逻辑需要把每一帧和它的所属的通道参数对应起来，保证ISP 使用的专属这一帧的寄存器参数集，等ISP 处理完后，软件需要把这一帧的ISP统计结果传递给3A算法的对应通道，保证3A 算法工作在正确的对象上。

2



知乎用户 回复 刘斯宁 (作者)

2020-02-18

非常感谢你的解答。

赞



知乎用户

2020-02-17

文中有提到ISP的两种处理方式，在线和离线，目前大部分ISP采用的是在线模式，HISI有几款ISP支持离线模式，他们这样设计的目的是什么？文中提到了在线模式的好处，没有说离线模式有哪些好处。

赞



刘斯宁 (作者) 回复 知乎用户

2020-02-18

在线模式用来处理实时数据流，比如通过MIPI协议进来的数据。离线模式可以处理任何介质进来的数据，比如通过USB、网络、SD卡读入内存的数据。在线模式需要设计专用的硬件通路，所以存在成本问题。离线模式从内存中读取数据，几乎没有额外成本。

2



知乎用户 回复 刘斯宁 (作者)

2020-02-18

一般芯片在FPGA仿真阶段都支持从DDR读取仿真pattern来验证数字逻辑电路的正确性，这样是不是可以说明其实大部分带有ISP的SOC都支持离线模式呢？只是通常大家都没有用起来。

赞

[查看全部 18 条回复](#)



首發於
無法抗拒的圖像技術

