

第3章 图像处理器算法研究设计

随着数码影像技术的发展,图像处理器的功能越来越完备。该章只是对图像处理器的几个基本功能进行的研究实现。共分八个小节分别对色彩插值、色彩校正、自动白平衡、伽马校正、色彩空间变换、平滑输出、运动图像对比度增强和自动聚焦的原理和实现方法加以阐述。这八个功能单元实现的功能分别是:

- (1) 色彩插值后产生了RGB像素数据。
- (2) 色彩校正是指对图像的色泽、阴影、色调、高亮区和颜色的饱和度进行调整,以弥补图像在各方面存在的缺陷。通过色彩校正后,图像传感器输出的图像,颜色会更鲜艳,更有层次感,更真实。
- (3) 伽马校正用于调节图像的亮度和对比度。如果图像没有进行Gamma校正,视觉上就会显得偏白或偏暗。
- (4) 色彩空间转换是将图像信息由RGB空间表示转换为YUV空间表示。具体实现按照公式即可。
- (5) 自动白平衡是针对图像传感器在不同环境下获取的图像会偏离原来的实际颜色进行校正,保证白色物体在任何环境中都保持为白色。算法实现上通过对当前帧图像信息的计算,反馈到输入端调节后续图像的白平衡。
- (6) 图像滤波的目的是降低图像噪声,提高信噪比。
- (7) 运动图像对比度增强用来提高图像的对比度,增强图像层次感。
- (8) 自动聚焦的功能是实现根据场景的变化随时聚焦。

下面分八个小节来具体说明其算法原理和实现方法。

3.1 插值(Color Interpolation)

3.1.1 插值算法的基本概念和几种典型插值算法介绍

不论是CCD还是CMOS图像传感器,为了简化工艺和降低成本,一个像素点往往只能记录从纯白到纯黑的系列色调,因而只能给出单色的色调值,不能同时给出RGB三组数据。因此,对于彩色的图像值的获取,这就需要借助于色彩滤镜阵列(CFA, Color Filter Array),即在图像传感器的像素表面覆盖一个多色的滤镜阵列。通过应用不同的色彩滤镜阵列,可以获得不同的图像输出阵列。其中,最常见的一种滤镜阵列的图像传感器获得的是一幅如图4所示的马赛克的图像阵列,即Bayer模型。

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

图 4 Bayer 图像阵列

显然，在这种图像阵列中，每个像素值只有一个颜色的色调值，另外两个颜色的色调必须利用相邻像素之间的相关性，通过数据计算获得，这些方法通常就被称为色彩算法^[3]。典型的几种插值算法如下：

(1) 最近临点法

对于二维图像，该法是“取待采样点周围4个相邻像素点中距离最近的1个邻点的灰度值作为该点的灰度值”。此算法虽然计算简单，但由于仅用对该采样点影响最大的(即最近的)像素的灰度值作为该点的值，而没有考虑其他相邻像素的影响(相关性)，因此重新采样后的图像灰度值有明显的 discontinuity，像质损失较大。

(2) 双线性内插法

作为对最近邻点法的一种改进，这种方法是“利用周围4个邻点的灰度值在两个方向上作线性内插以得到待采样点的灰度值”。即根据待采样点与相邻点的距离确定相应的权值计算出待采样点的灰度值。具体如下：

1) 红色/蓝色点处的像素的绿色分量插值等于其相邻的4个像素点的绿色分量的平均值。例如， $G8=(G3+G7+G9+G13)/4$ 。

2) 在绿色点处的红色/蓝色分量的插值

这分两种情况：一种情况是如果存在相邻的两个色素的红色/蓝色分量，就取红色/绿色分量的均值，例如 $B7=(B6+B8)/2$, $R7=(R2+R12)/2$ 。另一个情况是周围没有相邻的红色/蓝色分量，就取对角线方形的4个像素点处红色/蓝色分量的均值，例如：

$R8=(R2+R4+R12+R14)/4$, $B8=(B6+B8+B16+B18)/4$ 。

与最邻近法相比，双线性内插法由于考虑了待采样点周围四个直接邻点对待采样点的影响，因此基本克服了前者灰度不连续的缺点，但其代价是计算量有所增大。但进一步看，由于此方法仅考虑四个直接邻点灰度值的影响，而未考虑到各邻点间灰度值变化率的影响，因此具有低通滤波器的性质，使缩放后图像的高频分量受到损失，图像的轮廓变得较模糊。用此方法缩放后的图像与原图像相比，仍然存在由于计算模型考虑不周而产生的图像质量退化与精度降低的问题。

(3) 立方卷积法

立方卷积法是一个用邻近同色像素做三次插值，运算相对复杂。立方卷积法又分为一元立方法，二元立方法和 B 样条方法三种。

一元立方插值的插值核是一个一元三次方程如公式(3.1)所示：

$$h(x) = \begin{cases} (a+2)x^3 - (a+2)x^2 + 1 & 0 \leq x < 1 \\ ax^3 - 5ax^2 + 8ax - 4a & 1 \leq x < 2 \\ 0 & 2 \leq x \end{cases} \quad (3.1)$$

式中系数a有-1、0.5、0.75等几种选择。

二元立方插值的插值核是一个二元三次的方程如公式(3.2)所示：

$$h(x) = \begin{cases} [(-9b-6c+2)x^3 - (12b+6c-18)x + (-2b+6)]/6 & 0 \leq x < 1 \\ [(-b-6c)x^3 + (6b+30c)x^2 + (-12b-49c)x + (8b-24c)]/6 & 1 \leq x < 2 \\ 0 & 2 \leq x \end{cases} \quad (3.2)$$

式中系数(b,c)有(0.33,0.33)、(1.5,-0.25)等选择，与双线性内插法相比，立方卷积法不仅考虑了直接邻点的灰度值对待采样点权值的影响，还考虑了邻点间灰度值变化率的影响，因此后者所求得的待采样点灰度值更接近原(采样)值。此方法用进一步增大计算量来换取待采样点精度的进一步提高。

从前面几种算法的讨论不难看出：高阶算法有较好的插值质量，但运算量巨大；低阶算法运算量较小，但插值质量不是很理想。

3.1.2 本文用到的插值算法结构

经过研究一些典型的插值算法发现，引入相关性可以提高插值后重建图像的质量，而这种插值算法一般至少需要 5×5 的滤波器窗口，在硬件实现上就需要 4 行缓冲来保存图像数据，这样使得硬件开销很大，而本文所采用的算法只需 5×3 的滤波器窗口，也就是说只需要 2 行缓冲来保存缓冲数据，这便使硬件开销降低了。硬件则需要两个 RAM 做行缓冲，9 个寄存器构成一个 5×3 窗口，该模块的输出结果为窗口中间像素的 RGB 值，10 位位宽，高 2 位为符号位。具体实现的算法如下：

(1) 当 R 颜色成分位于窗口中心，四周为 G 颜色成分，此时插值情况如图 5 所示。

G_{11}	B_{12}	G_{13}	B_{14}	G_{15}
R_{21}	G_{22}	R_{23}	G_{24}	R_{25}
G_{31}	B_{32}	G_{33}	B_{34}	G_{35}

图 5 R 颜色成分位于窗口中心情况^[11]

这时为了提高重建图像质量，滤波器窗口中所有的绿色成分都参与了插值运算，其插值算法如公式(3.3)和(3.4)所示：

$$G'_{23} = \sum_{\substack{(g,h) \in \zeta \\ (m,n) \in \zeta}} g_1(G_{gh}, R_{mn}) \quad (3.3)$$

$$B'_{23} = G'_{23} - \sum_{\substack{(g,h) \in \zeta \\ (m,n) \in \zeta}} g_2(G_{gh}, B_{mn}) \quad (3.4)$$

(2) 当窗口中心为 G 颜色成分，上下为 B 颜色成分，左右为 R 颜色成分时插值情况如图 6 所示。

B ₁₁	G ₁₂	B ₁₃	G ₁₄	G ₁₅
G ₂₁	R ₂₂	G ₂₃	R ₂₄	R ₂₅
B ₃₁	G ₃₂	B ₃₃	G ₃₄	G ₃₅

图 6 G 颜色成分位于窗口中心情况^[11]

这时插值算法如公式(3.5)和(3.6)所示：

$$R'_{23} = G_{23} - \sum_{\substack{(g,h) \in \zeta \\ (m,n) \in \zeta}} f_1(G_{gh}, R_{mn}) \quad (3.5)$$

$$B'_{23} = G_{23} - \sum_{\substack{(g,h) \in \zeta \\ (m,n) \in \zeta}} f_2(G_{gh}, B_{mn}) \quad (3.6)$$

其中函数 f1 和 f2 是相关性函数，其他两种情况，插值算法与以上类似^[11]。

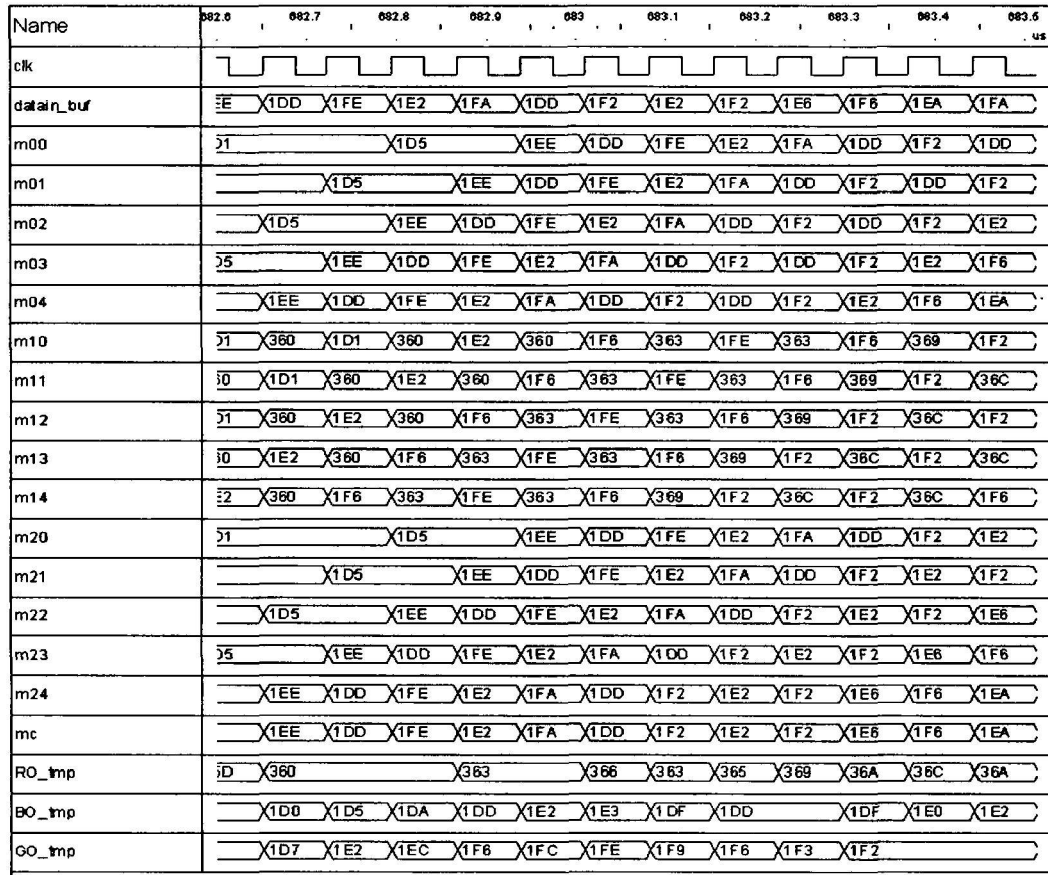


图 7 插值模块时序仿真

为了客观地比较图像的质量，可以使用峰值信噪比来作为评价指标^[12] R、G、B 三通

道不同的峰值信噪比分别如图 8、9、10 所示，峰值信噪比的计算方法如公式(3.7)所示：

$$PSNR = 10 \times \log_{10} \left[\frac{255^2}{\frac{1}{XY} \sum_{l=0}^{X-1} \sum_{m=0}^{Y-1} (o(l,m) - i(l,m))^2} \right] \quad (3.7)$$

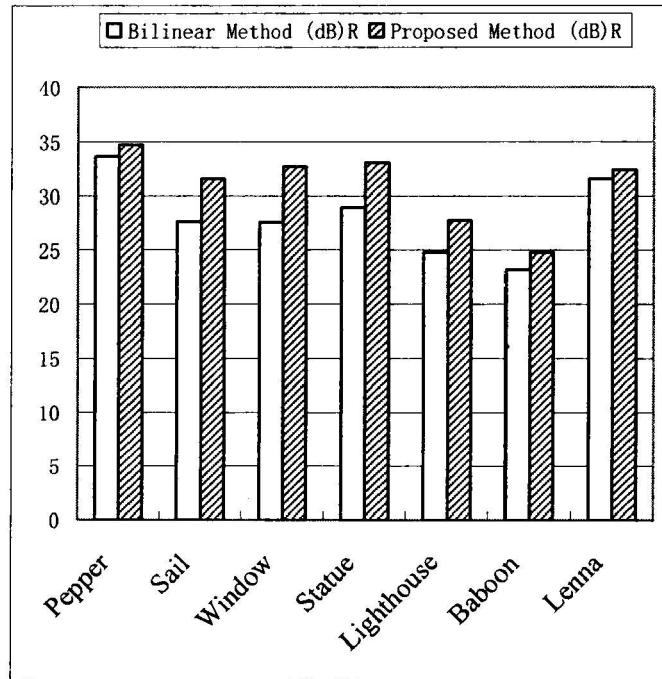


图 8 R 通道不同的 PSNR 比较图

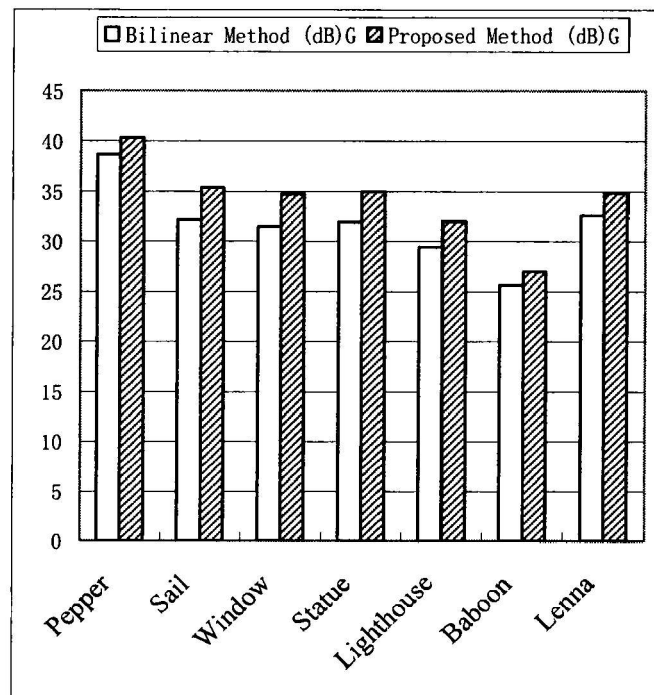


图 9 G 通道不同的 PSNR 比较图

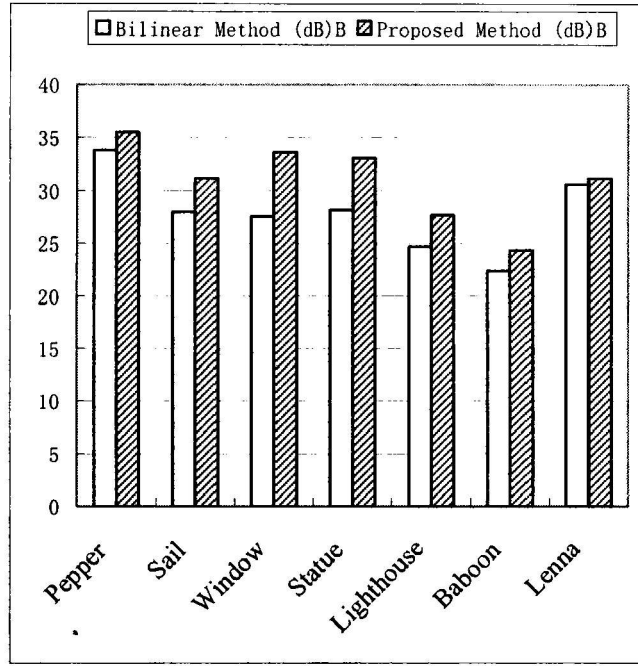


图 10 B 通道不同的 PSNR 比较图

用双线性插值和本文提出的新的插值方法进行比较，对于常用的测试标准图像测试后，可以看出新的插值方法有很大的优越性。

3.2 色彩校正

通过插值得到了 RGB 三原色齐全的图像，但传感器响应的这个图像与真实场景之间仍存在差异。这其中存在多方面的原因，涉及图像传感器中光学器件（棱镜）的光谱特性、场景的光源光照条件（诸如白光、荧光或者钨光）以及色彩滤镜的光谱特性等。

为了补偿这种差异，必须对图像的像素值（R、G、B）按式(3.8)进行变换处理。

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} +1.2840 & -0.1132 & 0.1197 \\ -0.1164 & +1.3908 & -0.1283 \\ -0.1390 & -0.1487 & +1.3550 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.8)$$

在硬件实现上，三个图像数据都取八位有效数据，这样修正后的矩阵表示如式(3.9)。

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} +1.28125 & -0.109375 & -0.11719 \\ -0.109375 & +1.390625 & -0.125 \\ -0.1328125 & -0.1484375 & +1.34375 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.9)$$

对于计算固定系数矩阵与输入矩阵的乘积，可以采用加法器和移位寄存器来实现系数相乘的运算，图 11 是 R 通道色彩校正算法结构图，其它两个通道算法与此类示。