

Autor:
Dipl. Ing. (FH) Uwe Furtner
Geschäftsführer BV-Produkte bei MATRIX Vision GmbH

Stand: 31.08.2001



Farbverarbeitung mit Bayer-Mosaic Sensoren

Einleitung

Im Bereich der industriellen Farbbildverarbeitung unterscheidet man die Dreichip-Kameras von den ‚normalen‘ Einchip-Kameras. Dreichip-Kameras haben 3 Grausensoren die mit unterschiedlichen optischen Filtern über ein Prisma pixelgenau auf das selbe Objekt schauen. Die einzelnen Rot-, Grün- und Blaufilter bedecken jeweils einen kompletten Sensorchip. Durch diese Methode erhält man die maximale Auflösung in jedem Farbauszug. Aus Kostengründen sind jedoch die Einchip-Farbkameras noch verbreiteter als die Dreichip-Kameras.

Einchip-Farbsensoren sind herkömmliche CCD Sensoren, die einen Farbfilter aufgebracht bekommen. Diese Filter bewirken, dass einzelne Pixel unterschiedlich auf die Wellenlängen des Lichtes reagieren. Da zu jedem Bildpunkt nur eine Farbkomponente gegeben ist, muss man um ein RGB-Pixel zu erhalten, die fehlenden Farbkomponenten ermitteln.

Den verbreitetsten Farbfilter bezeichnet man als ‚complementary color mosaic‘. Dabei wechseln sich pixelweise die Farben Cyan, Gelb, Magenta und Grün ab. Zu solchen CCD-Sensoren sind auch Chips erhältlich, die eine Umwandlung in RGB durchführen. Die typischen analogen Einchip-Farbkameras enthalten solche CCD-Sensoren.

Neuerdings hört man immer öfter den Begriff: ‚Bayer-Mosaic‘ oder ‚Primary Color Mosaic‘ - Sensoren. Hierbei handelt es sich auch um Einchip-Farbsensoren, die aber die Primärfarben Rot, Grün und Blau enthalten. In der digitalen Videotechnik kommen meist diese Sensoren zum Einsatz. Die Kamera liefert dann in der Regel auch nur die Originaldaten des Sensors und kein umgerechnetes RGB-Bild. Bei den immer schneller werdenden Kameras ist dies durchaus sinnvoll, um die Datenraten über einen PCI-Bus klein zu halten. Andererseits kann man jedoch den Wunsch des Anwenders verstehen, ein Farbbild von seiner Farbkamera zu sehen.

Genau hier soll dieser Artikel ansetzen – es wird erläutert wie ein Bayer-Mosaic-Filter aufgebaut ist und welche Konvertierungsmethoden zu RGB möglich sind.

Aufbau des Bayer-Mosaic Filters

Der Bayer Mosaic-Filter besteht aus einer RGB-Filtermaske, die auf einen Standard Grau Sensor aufgebracht wurde. Wie in Bild 1 zu sehen, wechseln sich in einer Zeile die Grünen und Blauen Pixel ab und in der nächsten Zeile sind es die Roten und Grünen. Für die nachfolgenden Berechnungsbeispiele sind die Pixel mit $p(x,y)$ durchnummeriert.

p_{00}	p_{10}	p_{20}	p_{30}	p_{40}
p_{01}	p_{11}	p_{21}	p_{31}	p_{41}
p_{02}	p_{12}	p_{22}	p_{32}	p_{42}
p_{03}	p_{13}	p_{23}	p_{33}	p_{43}
p_{04}	p_{14}	p_{24}	p_{34}	p_{44}

Bild 1: Aufbau des Bayer-Mosaic-Filters

Farbberechnung

Das Ziel ist es aus den Rohdaten des Bayer-Mosaic-Sensors ein Farbbild zu errechnen, das dem realen Objekt, bzw. einer Aufnahme mit einer Dreichip-CDD-Kamera möglichst nahe kommt.

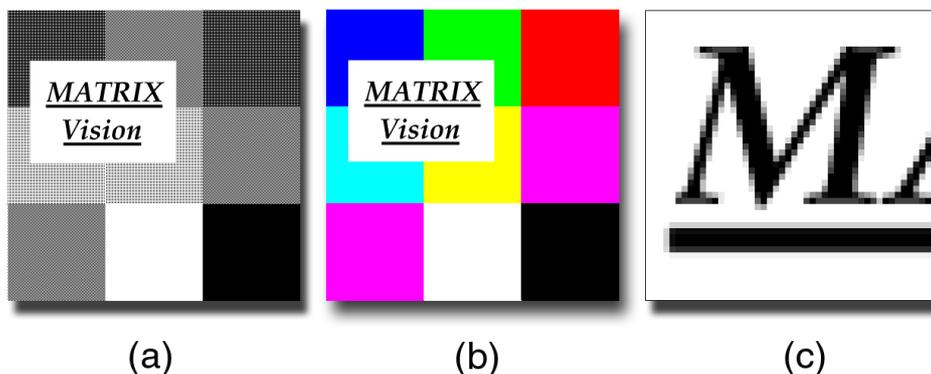


Bild 2: a) Bayer-Mosaic-Bild, b) ideales Umrechnungsergebnis, c) Bildausschnitt

Bild 2a zeigt die Rohdaten des Bayer-Mosaic-Sensors als Graubild dargestellt. Bild 2b zeigt das ideale Farbbild, wie es beispielsweise von einer Dreichip-Kamera stammen könnte. Im Bildausschnitt 2c sind keine Farbartefakte zu sehen.

Da nur in jeder zweiten Spalte und Zeile des Bayer-Mosaics beispielsweise der Blauwert eines Pixels genau bekannt ist, müssen die Werte dieser Farbkomponente in den umgebenden Pixel berechnet werden.

Hierzu gibt es verschiedene Methoden, die im folgenden vorgestellt werden.

Einfachste Berechnungsmethode

Die allereinfachste Methode ist die Verwendung eines 2x2 Operators. Dieser Operator läuft pixelweise durch das Bild. Der Operator wird bei einem Zeilenwechsel auf die nächste Pixelreihe gesetzt. Die Berechnungsvorschrift für jede Operatorposition lautet: Nehme die vorhandenen Farbkomponenten ohne sie mit Ihren Nachbarpixel zu verrechnen.

$$\begin{aligned} R_{11} &= p_{21} \\ G_{11} &= p_{11} \quad \{ \text{oder } (p_{11} + p_{22})/2 \} \\ B_{11} &= p_{12} \end{aligned}$$

$$\begin{aligned} R_{21} &= p_{21} \\ G_{21} &= p_{31} \quad \{ \text{oder } (p_{31} + p_{22})/2 \} \\ B_{21} &= p_{32} \end{aligned}$$

....

$$\begin{aligned} R_{12} &= p_{23} \\ G_{12} &= p_{13} \quad \{ \text{oder } (p_{13} + p_{22})/2 \} \\ B_{12} &= p_{12} \end{aligned}$$

Bei dieser Methode werden die blau bzw. rot gefärbten Pixel einfach vervierfacht und die grünen Pixel verdoppelt. Betrachtet man den Rot- oder Blauauszug (Bild 3c) des Ergebnisbildes sieht man dass immer 4 benachbarte Pixel den gleichen Farbwert haben.

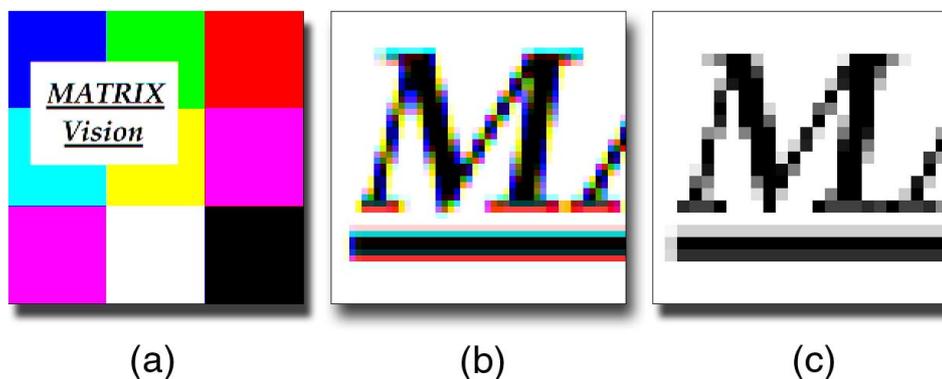


Bild 3: Ergebnis der einfachen Methode, a) Testbild komplett, b) Ausschnitt, c) Blauauszug

Man geht bei dieser Methode davon aus, dass die fehlende Farbkomponente an einer Pixelposition der Farbkomponente seines Nachbarn entspricht, was natürlich bei realen Farbverläufen nicht der Fall ist.

Schnelle Berechnung für eine Livedarstellung

Für die Darstellung eines Farbbildes im Livemode wird ein Verfahren gesucht, das möglichst wenig Rechenaufwand benötigt um eine hohe Refreshrate des Livebildes zu erreichen. Das obige Verfahren wäre bezüglich des geringen Rechenaufwandes für die Livedarstellung geeignet. Nur sollten wir das Verfahren geringfügig verändern um eine wesentlich bessere Wirkung zu erzielen. Mit der ‚Einfachsten Methode‘ werden die Rot- und Blaukomponenten vervierfacht. Es ist sicherlich nicht notwendig zunächst die 4fache Datenmenge zu erzeugen um diese dann darzustellen.

Betrachtet man lediglich die blauen Pixel und lässt die Zwischenpixel weg, so erhält man ein Bild des Blaukanals. Ebenso kann man auch mit Rot und Grün verfahren. Legt man diese resultierenden Farbebenen übereinander erhält man ein Farbbild mit der halben horizontalen und vertikalen Auflösung des Originalsensorbildes. Bei diesem Bild sind die einzelnen Farbauszüge um jeweils einen halben Pixel in horizontaler und vertikaler Richtung gegenüber der Realität verschoben. Skaliert man bei der Darstellung dieses Bild um den Faktor 2 über eine bilineare Interpolation, so erhält man ein wesentlich besseres visuelles Ergebnis als bei der ‚Einfachsten Methode‘. Es sind keine Farblöcke mehr zu sehen.

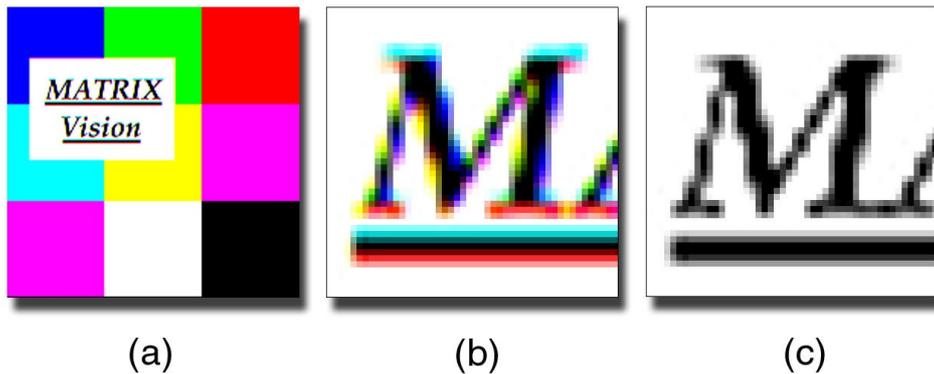


Bild 4: Ergebnis der schnellen Methode, a) Testbild komplett, b) Ausschnitt, c) Blauauszug

Das Berechnungsbeispiel für dieses Verfahren:

$$\begin{aligned} R00 &= p01 \\ G00 &= p00 \quad \{ \text{oder } (p00 + p11)/2 \} \\ B00 &= p10 \end{aligned}$$

$$\begin{aligned} R10 &= p21 \\ G10 &= p20 \quad \{ \text{oder } (p20 + p31)/2 \} \\ B10 &= p30 \end{aligned}$$

....

$$\begin{aligned} R01 &= p03 \\ G01 &= p02 \quad \{ \text{oder } (p02 + p13)/2 \} \\ B01 &= p12 \end{aligned}$$

Allgemeine Interpolation

Um die volle Farbbildauflösung zu erreichen, muss die örtliche Verschiebung der Farbkomponenten berücksichtigt werden. Dabei geht man davon aus, dass sich die fehlenden Farben als Mittelung der benachbarten Pixel der gleichen Farbkomponente ergeben. Für fließende Farbverläufe stimmt diese Annahme sicherlich in erster Näherung. Dabei werden immer die nächstliegenden Farbkomponenten mit einem 3x3 Operator gemittelt:

$$\begin{aligned} R11 &= (p01+p21)/2 \\ G11 &= p11^* \\ B11 &= (p10+p12)/2 \end{aligned}$$

$$\begin{aligned}R21 &= p21 \\G21 &= (p20+p31+p22+p11)/4 \\B21 &= (p10+p30+p12+p32)/4 \\&\dots\end{aligned}$$

$$\begin{aligned}R12 &= (p01+p21+p03+p23)/4 \\G12 &= (p11+p22+p13+p02)/4 \\B12 &= p12\end{aligned}$$

$$\begin{aligned}R22 &= (p21+p23)/2 \\G22 &= p22^*) \\B22 &= (p12+p32)/2\end{aligned}$$

Das Berechnungsergebnis ist in Bild 5 zu sehen.

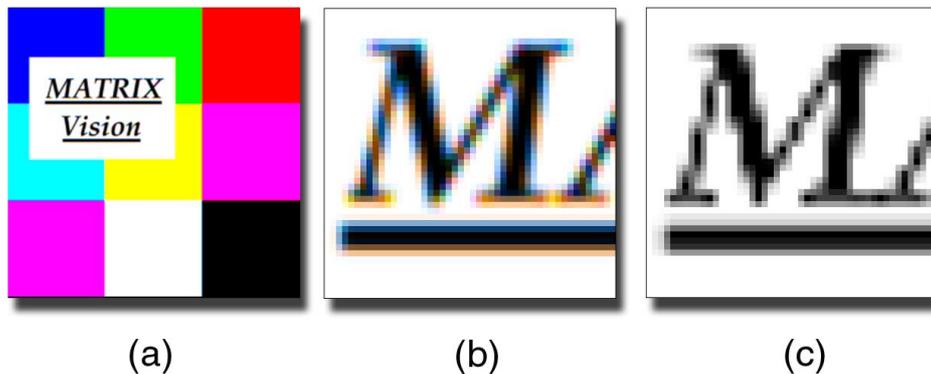


Bild 5: Ergebnis der allgemeinen Interpolation, a) Testbild komplett, b) Ausschnitt, c) Blauauszug

*) Laut Sensorhersteller kann es einen Helligkeitsunterschied zwischen einem Grünpixel innerhalb der blauen Zeile gegenüber eines Grünpixels aus der Rotzeile geben. Deshalb sollten die Grünpixel nicht direkt verwendet werden, sondern entweder mit Nachbargrünwerten gemittelt oder über einen Faktor korrigiert werden. Dieser Korrekturfaktor schwankt aber von Sensor zu Sensor und sollte im Rahmen eines Weissabgleichs detektiert werden.

Schärfende Interpolation

Bei der allgemeinen Interpolation geht man davon aus, dass sich die Farbe mit der halben Frequenz gegenüber der Intensität (Luminanz) verändert. Dies stimmt sicherlich für einen Einchip-Farbsensor, jedoch kommen wir damit mit unserem Ziel nicht weiter, möglichst nahe an die Farbauflösung einer Dreichip-Kamera heranzukommen. An Objektkanten entstehen Farbartefakte, die bei der automatischen Farbbildauswertung sehr stören können. Außerdem erscheint durch die Interpolation das resultierende Bild etwas unscharf. Die Artefakte kann man verringern und die Bildschärfe gleichzeitig erhöhen, wenn man die Kanteninformationen mit betrachtet.

In [5] wird gezeigt, wie man durch die Untersuchung der Gradienten in unterschiedlichen Richtungen eines Pixels, Wichtungsfaktoren für die Nachbapixel ermittelt. Dieses Verfahren ist natürlich noch rechenaufwendiger als die allgemeine Interpolation.

Hardwarelösung

Bislang sind wir immer davon ausgegangen, dass die Rohdaten des Sensors im PC Bildspeicher landen und wir diese mit entsprechenden Algorithmen in RGB umrechnen müssen. Diese Umrechnung kann auch direkt von einem Frame Grabber erledigt werden, was die MVtitan-DIG der Firma MATRIX Vision zeigt. Damit ist der PC nicht mit der Umrechnung belastet. Die Karte verwendet die allgemeine Interpolation auch für den Livemode wodurch keine Qualitätsunterschiede zwischen Live- und Snapmode zu sehen sind.

Fazit

Der zu verwendende Algorithmus ist abhängig vom Anwendungsfall. Mit viel Aufwand kann man das Bild ‚schön-rechnen‘. Natürlich muss man sich immer vergegenwärtigen, dass durch die Farbumrechnung die zwischenliegenden Farbwerte abgeschätzt und eigentlich erfunden werden. Nur bei Dreichip-Kameras erhält man wirklich die maximale Auflösung. Vergleicht man aber digitale Bayer-Mosaic-Kameras mit Standard Einchip-Farbkameras so ist die Bildqualität weitaus besser. Damit ergibt sich eine Qualitätsreihenfolge, die sich in ähnlicher Form auch im Preis der Kameras widerspiegelt.

Wenn die Visualisierung des Kamerabildes zweitrangig ist, kann man auf die komplette Umrechnung des Bildes verzichten und hat damit nur 1/3 der Datenmenge. Stattdessen kann man lediglich die Auswertebereiche umrechnen oder die Bildverarbeitungsalgorithmen können direkt mit einem Bayer-Mosaic-Bild umgehen.

Literaturhinweise

[1] Brainard, D., "Bayesian Method for Reconstructing Color Images from Trichromatic Samples", IS&T, 47th Annual Conference, pp. 375-380, 1994.

[2] Cok, D., "Reconstruction of CCD Images Using Template Matching", IS&T, 47th Annual Conference, pp.380-385, 1994.

[3] Gann, R., Reviewing and Testing Desktop Scanners. Hewlett-Packard Company, 1994.

[4] Wandell, Brian A., Foundations of Vision. Sinauer, Sunderland, Mass., ch.2-4,8, 1995.

[5] Ed Chang, Shiufun Cheung, Davis Pan, „Color Filter Array Recovery Using a Threshold-based Variable Number of Gradients“, IS&T/SPIE Conference on Sensors, 1999.