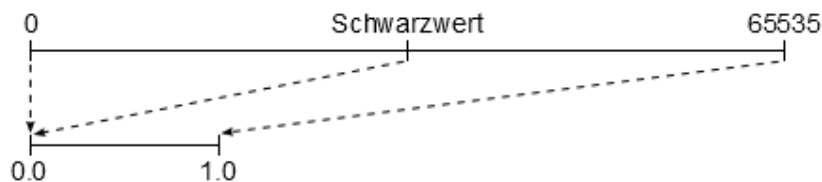


## Aufgabe 2.1: Kamera Pipeline (7 Punkte)

Bei dieser Aufgabe geht es darum, Teile der automatischen Verarbeitungsschritte, die nach der Bildaufnahme mit einem digitalen Sensor angewendet werden, zu implementieren.

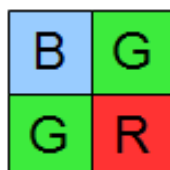
Die Bilder `IMG_1.tiff` bis `IMG_3.tiff` sind als TIFF abgespeicherte RAW-Bilder, die sowohl die bei der Bildaufnahme gesammelte Informationen im Header enthalten sowie die originalen aus dem Sensor ausgelesenen Pixelwerte. Führen Sie folgende Schritte aus, um aus dem RAW-Bild ein digitales Farbbild zu erstellen:

- Lesen Sie zuerst die relevanten Informationen aus dem Header aus. Verwenden Sie dazu die Funktion `imfinfo`. In unserem vereinfachten Beispiel sind nur die Werte `BlackLevel` und `AsShotNeutral` interessant. `BlackLevel` bezeichnet den Schwarzwert der Kamera, welcher aufgrund von Rauschen nicht bei 0 liegt. `AsShotNeutral` bezeichnet den Farbwert von "Weiß", der für das White Balancing verwendet werden kann.
- Bei den Bildern sind die Pixelwerte mit 16 Bit kodiert. Verschieben und skalieren Sie die Pixelwerte so, dass Schwarz (`BlackLevel` und Werte darunter) auf 0 und Weiß auf 1 abgebildet wird. Dazu soll das Eingabebild vom Intervall  $[\text{BlackLevel}, 65535]$  auf das Intervall  $[0.0, 1.0]$  abgebildet werden, wie in folgender Abbildung verdeutlicht:

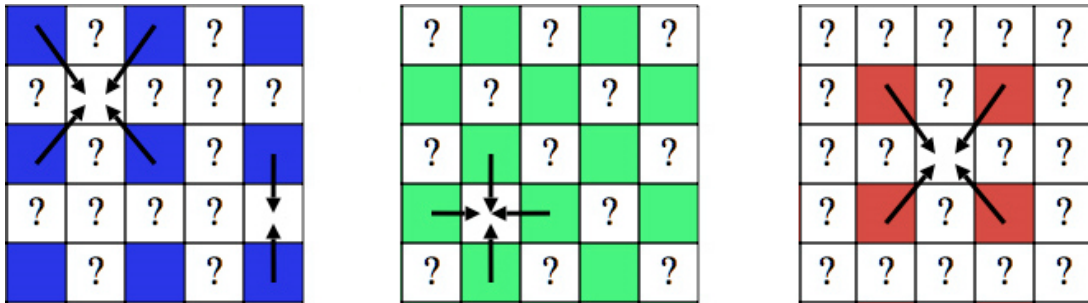


In anderen Worten: Der Schwarzwert soll 0.0 und der Wert 65535 (Weiß) soll 1.0 werden. Alle Werte zwischen `BlackLevel` und 65535 sollen zwischen 0.0 und 1.0 liegen und alle Werte kleiner dem Schwarzwert müssen auch auf 0.0 gesetzt werden.

- Führen Sie ein Demosaicing durch, um für alle Pixelpositionen RGB Farbwerte zu bekommen. Der Bildsensor ist mit einem Farbfiler (Bayer Pixel Pattern) überzogen, der nur eine bestimmte Farbkomponente zum Sensor durchlässt. Diese Filter sind in 2x2 Pixelmustern angeordnet. Da die Sensitivität des menschlichen Auges im grünen Bereich am besten ist, wird der grüne Filter auf zwei der vier Sensoren aufgetragen. Verschiedene Kameramodelle verwenden verschiedene Muster, in unserem Fall wurde folgender Filter verwendet:



Dieses Muster wird beginnend vom linken, oberen Bildpunkt über dem ganze Bild wiederholt. Um alle drei Farbkanäle für jeden Pixel zu bekommen, müssen an den Stellen, an denen Informationen fehlen, die benachbarten Werte interpoliert werden, wie auf der folgenden Abbildung verdeutlicht:



Um die fehlenden Blau- und Rotwerte zu berechnen wird das Mittel der angrenzenden 2 oder 4 Werte bestimmt. Für die grünen Werte müssen jeweils die vier Nachbarn gemittelt werden. Nachdem alle 3 Kanäle über das ganze Bild bestimmt wurden, fügen Sie alles zu einem einzigen RGB-Bild zusammen. Hinweis: Der `demosaic` Befehl in Matlab verwendet ein komplizierteres Interpolationsverfahren. Er kann als Vorschau helfen, darf jedoch in der Lösung nicht verwendet werden.

**Vorgehensweise:** Erstellen Sie sich zuerst Matrizen für den Rot-, Grün- und Blaukanal und füllen Sie diese mit den entsprechenden Werten aus dem Bild `I`. Für den Rotkanal funktioniert das folgendermaßen:

```
bayer = [3 2;2 1];
bayer = repmat(bayer, size(I,1)/2, size(I,2)/2);

R = zeros(size(I));
R(bayer==1) = I(bayer==1);
```

Anschließend müssen noch die fehlenden Werte interpoliert werden. Verwenden Sie dazu den Befehl `imfilter` mit folgenden Filtern:

|      |     |      |
|------|-----|------|
| 0.25 | 0.5 | 0.25 |
| 0.5  | 1   | 0.5  |
| 0.25 | 0.5 | 0.25 |

Filter für Rot- und Blaukanal.

|      |      |      |
|------|------|------|
| 0    | 0.25 | 0    |
| 0.25 | 1    | 0.25 |
| 0    | 0.25 | 0    |

Filter für Grünkanal.

- Führen Sie nun den Weißabgleich mithilfe des als "Weiß" bestimmten Farbwerts `AsShotNeutral` durch. Dafür müssen Sie lediglich jeden Farbkanal durch den jeweiligen Wert in `AsShotNeutral` dividieren. Da dabei auch Werte größer als 1 entstehen können, müssen diese noch anschließend auf 1 gesetzt werden.

- Die Bilder sind nach dem vorigen Schritt in der Regel noch zu dunkel. Erhöhen Sie den Kontrast der Bilder mittels dem Befehl `imadjust`. Finden Sie dafür für jedes Bild eigene Parameter, um ein möglichst anspruchsvolles Bild zu erzeugen.

Zeigen sie für ein Bild alle Schritte im PDF-Report und interpretieren Sie diese. Zeigen Sie auch alle finalen Farbbilder. Welche Werte habe Sie für `imadjust` verwendet und warum?