

# Bildüberlagerung mit Hilfe von Transparenz

Von Steve Eddins

Wenn man sich zwei Bilder gleichzeitig ansehen will – etwa um das Ursprungsbild und Resultat einer bestimmten Bildbearbeitungs-Operation zu vergleichen oder um verschiedene Bilder der gleichen Szenerie zu vergleichen – dann ist es oft effizienter, beide Bilder übereinander zu legen, anstatt sie nebeneinander zu betrachten. In einem Handle Graphics®-Bildobjekt kann man mit der ‚AlphaData‘-Eigenschaft von Bildern jedem Pixel einen individuellen Transparenzgrad zuweisen<sup>1</sup>.

Dieser Artikel stellt zwei Transparenzmuster vor, die auf ein breites Spektrum an Bilddarstellungen angewendet werden können: Ein Schachbrett, das abwechselnd Teile beider Bilder durchscheinen lässt, sowie ein bilddatenabhängiges Muster, das den Deckungsgrad einer vorausgewählten Farbe variiert.

## Schachbrett-Transparenz

Mit dieser Methode werden im Beispiel das Ursprungsbild und Resultat einer Graustufenkonvertierung verglichen. Im ersten Schritt werden dazu beide Bilder gleichzeitig angezeigt (Abb. 1).

```
rgb = imread('peppers.png');
imshow(rgb);
I = rgb2gray(rgb);
hold on
% Save the handle for later use
h = imshow(I);
hold off
```

Natürlich ist auf diese Weise nur das Graustufenbild sichtbar. Dieses wird nun mit einem Schachbrett-Transparenzmuster mit entsprechend undurchsichtigen bzw. transparenten Pixelbereichen versehen. (Abb. 2).

```
[M,N] = size(I);
block_size = 50;
P = ceil(M / block_size);
Q = ceil(N / block_size);
alpha = checkerboard(block_size, ...
    P, Q) > 0;
alpha = alpha(1:M, 1:N);
set(h, 'AlphaData', alpha);
```



ABB. 1: Graustufenbild mit darunterliegendem Farbbild.



ABB. 2: Das Graustufenbild mit Schachbrett-Transparenzmuster lässt Teile des Farbbildes durchscheinen.

<sup>1</sup> ‚AlphaData‘ ist eine  $M \times N$ -Matrix identischer Größe wie das Bild, deren innerhalb des Wertebereichs  $[0 \ 1]$  liegende Elemente jeweils den Deckungsgrad eines Pixels definieren.



ABB. 3: DHM-Bild von Peppercorn Hill und North Pond, Massachusetts. Helle Pixel entsprechen hochgelegenen Geländepunkten (Daten mit freundlicher Genehmigung der U.S. Geological Survey)



ABB. 4: Die Influence Map macht das bergab fließende Wasser sichtbar.



ABB. 5: Das DHM-Bild mit darübergelegtem Grünbild.

Jetzt sind Anteile beider Bilder sichtbar und man kann Ursprungsbild und Graustufenbild auf einfache Weise vergleichen.

### Bilddatenabhängige Transparenz

Das zweite Beispiel eines Transparenzmusters ist kreativer. Bei dieser Methode färbt man einige Pixel des einen Bildes in Abhängigkeit von den Pixelwerten im zweiten Bild ein. In diesem Beispiel soll in einem Digitalen Höhenmodell (DHM) durch Überlagerung der Weg von bergab fließendem Wasser sichtbar gemacht werden (Abb. 3).

```
E = imread('peppercorn_hill.png');
imshow(E, 'InitialMag', 'fit')
```

Dazu erzeugt man zunächst eine sogenannte Influence Map, die den Verlauf des Wasserflusses hervorhebt, beginnend am Gipfel des Peppercorn Hill (Abb.4).

```
I = imread('influence_map.png');
imshow(I, 'InitialMag', 'fit')
```

Losgelöst aus dem Kontext des DHM-Bildes lässt sich diese Influence Map nur schwer interpretieren – man erkennt beispielsweise nur mit Mühe, wo genau der

Fluss den North Pond erreicht. Um beide Bilder gleichzeitig anzuzeigen, macht man folgendes:

1. Das ursprüngliche DHM-Bild anzeigen.
2. Ein komplett grünes Bild über das DHM-Bild legen (Abb. 5).

```
imshow(E, 'InitialMag', 'fit')
% Make a truecolor all-green image.
green = cat(3, zeros(size(E)), ...
    ones(size(E)), zeros(size(E)));
hold on
h = imshow(green);
hold off
```

3. Die Transparenz sämtlicher Pixel des grünen Bildes mithilfe der Pixel der Influence Map regulieren (Abb. 6).

```
% Use our influence map as the
% AlphaData for the solid green image.
set(h, 'AlphaData', I)
```

Nun ist deutlich zu erkennen, wie das Wasser vom Gipfel herab in den Teich fließt und diesen am Südende wieder verlässt. ■

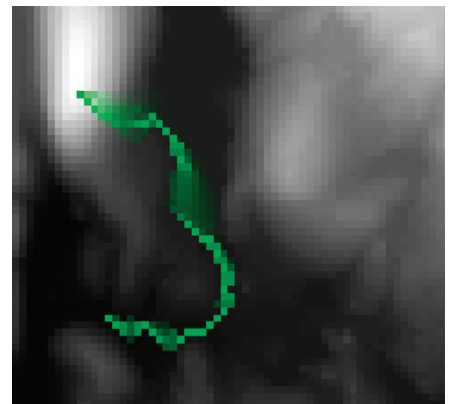


ABB. 6: DHM-Bild mit überlagerter Bild des Wasserflusses.

### Quellen

**TRANSPARENZ IN MATLAB®**  
[www.mathworks.de/nn9/transparency](http://www.mathworks.de/nn9/transparency)

**PRODUKTE FÜR DIE BILDVERARBEITUNG**  
[www.mathworks.de/nn9/imageprocessing](http://www.mathworks.de/nn9/imageprocessing)