

Bundeswettbewerb Informatik: Aufgabe 5

Lösungsidee

Zum Herausfinden des fehlerhaften Bereichs vergleiche ich die Pixel des hoch aufgelösten Bildes mit den Pixeln des (dazugehörigen) niedrig aufgelösten Bildes. Beim Vergleich wird das Auflösungsverhältnis zwischen beiden Bildern beachtet, denn einem Pixel im niedrig aufgelösten Bild entsprechen mehreren Pixeln im hoch aufgelösten Bild.

$$\text{Auflösungsverhältnis} \approx \frac{H}{h} \approx \frac{B}{b}$$

H: Höhe des hoch aufgelösten Bildes
h: Höhe des niedrig aufgelösten Bildes
B: Breite des hoch aufgelösten Bildes
b: Breite des niedrig aufgelösten Bildes

Beim Vergleich der Pixel kommt es vor, dass Pixel als fehlerhaft gekennzeichnet werden, die gar nicht fehlerhaft sind. Diese nicht-fehlerhaften Pixel müssen herausgefiltert und entfernt werden.

Hat man den Fehler im hoch aufgelösten Bild gefunden muss man diesen nur noch mit Hilfe des niedrig aufgelösten Bildes beheben. Dabei werden mehrere beschädigte Pixel durch einen Pixel des niedrig aufgelösten Bildes ersetzt (je nach Auflösungsverhältnis). Man skaliert das niedrig aufgelöste Bild also auf die Größe des hoch aufgelösten Bildes und ersetzt nur die fehlerhaften Pixel mit den Pixeln des hoch skalierten Bildes.

Programm-Dokumentation

Die Lösungsidee ist von mir in der Programmiersprache C umgesetzt. Zur Lösung der Aufgabe benötigte ich die Strukturen **pixel** und **ppm**. In die **ppm**-Struktur kann man alle nötigen Informationen einer ppm-Datei speichern, so z.B. den Dateinamen, die Identifikation (Bitmap-Typ), die Höhe und Breite des Bildes und die Pixeldaten. Diese Pixeldaten sind als ein 2-dimensionales Array bestehend aus der Struktur **pixel** in der Struktur **ppm** enthalten. Eine Struktur **pixel** besteht aus 3 **unsigned-char**-Variablen, die den RGB-Wert eines Pixels im Bild repräsentieren.

In der **main**-Funktion lasse ich zunächst beide Bilder (Bild mit dem Fleck und niedrig aufgelöstes Bild) in zwei **ppm**-Strukturen umwandeln. Dann wird der fehlerhafte Bereich im hoch aufgelösten Bild mit Hilfe der Funktion **fehlerhafter_bereich** herausgefunden und abgespeichert.

Die Funktion **fehlerhafter_bereich** findet fehlerhafte Pixel, indem die Pixel des hochauflösten Bildes mit denen des niedrig aufgelösten Bildes verglichen werden. Dabei muss das Auflösungsverhältnis, welches die Funktion **aufloesungsverhaeltnis** herausfindet, zwischen den beiden Bildern beachtet werden. Ist das Auflösungsverhältnis z.B. 3, dann werden 9 Pixel (3x3) des hoch aufgelösten Bildes mit einem Pixel des niedrig aufgelösten Bildes verglichen. Sind die verglichenen Pixel ähnlich, so werden diese nicht als fehlerhaft markiert. Ansonsten werden sie als fehlerhaft markiert und in eine **ppm**-Struktur abgespeichert.

Das Vergleichen zweier Pixel übernimmt die Funktion **vergleiche_pixel**. Beim Vergleichen der Pixel werden gewisse Toleranzen zugelassen. Je höher die Toleranz, desto eher wird ein Pixel als „ähnlich“ erkannt und somit nicht als fehlerhaft gekennzeichnet.

Es kommt vor, dass Pixel fehlerhaft gekennzeichnet werden, die gar nicht zum Fleck gehören. Zum

herausfiltern des Flecks gibt es die Funktion **nur_bereich**. Diese Funktion entfernt alle Pixel, die nicht zu einem größeren zusammenhängenden Bereich gehören.

Die Funktion **fehlerhafter_bereich** gibt den Zeiger auf die **ppm**-Struktur, die den Fleck enthält, an die **main**-Funktion zurück. Dort wird der Zeiger dann der Funktion **schreibe_ppm_in_datei** übergeben, die die **ppm**-Struktur als Bild abspeichert.

Als nächstes wird die Funktion **korrigiere** aufgerufen, die die **ppm**-Strukturen des hoch und des niedrig aufgelösten Bildes, sowie die **ppm**-Struktur, die den fehlerhaften Bereich enthält, als Parameter übergeben bekommt.

Es werden nun fehlerhafte Pixel aus dem hochaufgelösten Bild mit Pixeln des niedrig aufgelösten Bildes ersetzt. Mit Hilfe der **ppm**-Struktur, die den fehlerhaften Bereich enthält, erfährt die Funktion **korrigiere** welche Pixel fehlerhaft sind und ersetzt werden müssen. Der Zeiger auf die **ppm**-Struktur, welche das korrigierte **ppm**-Bild enthält, wird an die **main**-Funktion zurückgegeben und dann mit Hilfe der Funktion **schreibe_ppm_in_datei** als Bild abgespeichert.

Programm-Ablaufprotokoll

Beim Ausführen des Programms mit falschen Parametern oder bei der Option **-h** wird folgende Hilfe ausgegeben:

```
$ ./aufgabe5 -h
Benutzung: ./aufgabe5 [Optionen]
Optionen:
-f <Dateiname>   Dateiname des hoch aufgelösten Bildes mit dem
                  Fleck
-n <Dateiname>   Dateiname des niedrig aufgelösten Bildes
-o <Dateiname>   Ausgabe: Dateiname des Bildes mit dem Fleck
-k <Dateiname>   Ausgabe: Dateiname des korrigierten Bildes
-h              Ausgeben dieser Hilfe
```

Wird das Programm ohne Parameter aufgerufen, so wird das Bild **bild1-fleck.ppm** mit Hilfe des Bildes **bild1-klein.ppm** korrigiert. Der Fleck wird dann in die Datei **fleck.ppm** und das rekonstruierte Bild in die Datei **korrigiert.ppm** abgespeichert. Es erscheint folgendes auf dem Bildschirm:

```
$ ./aufgabe5
Der fehlerhafte Bereich im Bild bild1-fleck.ppm wurde mit dem
Bild bild1-klein.ppm korrigiert.
Der fehlerhafte Bereich wurde in der Datei fleck.ppm
abgespeichert.
Das korrigierte Bild wurde in der Datei korrigiert.ppm
abgespeichert.
```

Möchte man das andere Bild korrigieren, so gibt man folgende Parameter an:

```
$ ./aufgabe5 -f bild2-fleck.ppm -n bild2-klein.ppm -o \
fleck2.ppm -k korrigiert2.ppm
Der fehlerhafte Bereich im Bild bild2-fleck.ppm wurde mit dem
Bild bild2-klein.ppm korrigiert
Der fehlerhafte Bereich wurde in der Datei fleck2.ppm
abgespeichert
Das korrigierte Bild wurde in der Datei korrigiert2.ppm
abgespeichert
```

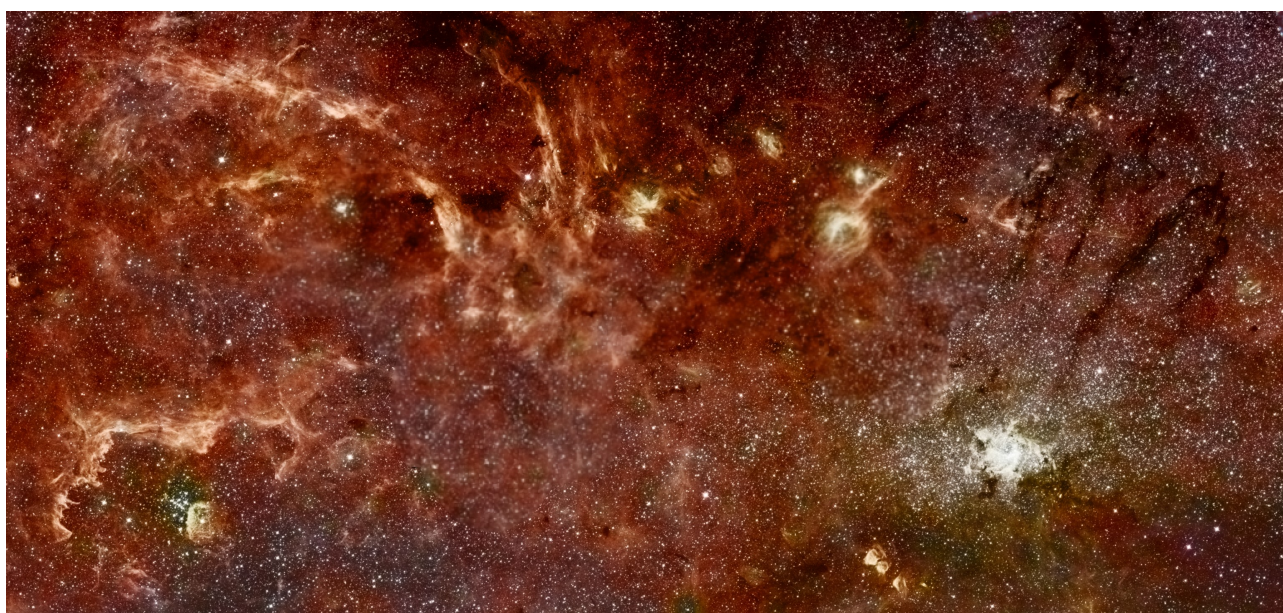
Mit der Option **-o** wurde der Fleck nun in der Datei **fleck2.ppm** abgespeichert und durch die Option **-k** wurde das rekonstruierte Bild in die Datei **korrigiert2.ppm** abgespeichert.

Um die Dateinamen so angeben zu können müssen die Bilder auch in dem Ordner sein, wo das Programm ist.

Die Ausgabebilder sind auf den nächsten Seiten (verkleinert) abgedruckt.



fleck.ppm



korrigiert.ppm



fleck2.ppm



korrigiert2.ppm

Zur Qualität der Bildrekonstruktion

Gerade im letzten Bild (**korrigiert2.ppm**) erkennt man nicht wirklich, dass das Bild je einen Fehler hatte. Vielleicht ist dies auch dadurch bedingt, dass dieser Teil der Galaxie in Wirklichkeit vielleicht ein bisschen „neblig“ ist.

Ist der Fleck nicht so groß wie im Bild **bild1-fleck.ppm** ist es durchaus möglich diese Art der Bildrekonstruktion anzuwenden.

Der Nachteil meiner Methode der Bildrekonstruktion ist, dass dabei wichtige Details verloren gehen. Denn in dem Bereich wo der Fleck ist wurde die Auflösung im hoch aufgelösten Bild verringert. Für wissenschaftliche Zwecke ist diese Methode also nicht geeignet, denn dort werden die Details benötigt.

Ein anderer Ansatz den fehlerhaften Bereich zu korrigieren wäre gewesen den dunklen Bereich mit Hilfe des niedrig aufgelösten Bildes aufzuhellen. Dabei bliebe die hohe Auflösung des Bildes erhalten.