

Bildverarbeitung

Prof. Schwarz

Astronomische Bildverarbeitung von Aufnahmen der Whirlpool-Galaxie mit Matlab

Lorenz Saalmann (5391121)

SS 25

1 Einführung

Die Aufnahme von Bildern astronomischer Objekte von der Erde aus ist nur unter bestimmten Bedingungen möglich. Während Sterne bereits mit dem bloßen Auge beobachtet werden können, erfordert die Auflösung von Galaxien wie der Whirlpool-Galaxie die Verwendung von Teleskopen und speziellen Bildverarbeitungsmethoden. Das Hauptproblem einer solchen Aufnahme ist es, die sehr geringe Lichtmenge dieser Objekte zu erfassen und von Rauschen zu unterscheiden. Rauschen wird durch verschiedene Faktoren verursacht, wie atmosphärische Störungen, thermisches Rauschen in den Sensoren oder Lichtverschmutzung. Längere Belichtungszeiten verbessern das Signal-zu-Rauschverhältnis einer Aufnahme, sind jedoch aufgrund der Erdrotation nur begrenzt einsetzbar, da das Bild sonst verwischt. Um dieses Problem zu umgehen, werden viele Aufnahmen desselben Objekts nacheinander aufgenommen, um aus diesen Daten ein rauscharmes Signal zu erhalten[1]. In dieser Ausarbeitung wird die Verarbeitung solcher Rohaufnahmen der Whirlpool-Galaxie mit Matlab zu einem einzelnen Bild beschrieben. Dafür werden die einzelnen Aufnahmen zunächst vorverarbeitet, bevor sie ausgerichtet und gestapelt werden. Abschließend wird das Ergebnis nachbearbeitet und ausgewertet.

2 Die Aufnahme

Die Whirlpool-Galaxie, auch bekannt als Messier 51a, ist eine Spiralgalaxie. Sie liegt im Sternbild der Jagdhunde und ist etwa 24 Millionen Lichtjahre von der Milchstraße entfernt. Bei dem leuchtenden Nebel handelt es sich vor allem um Wasserstoff, der sowohl Licht von Sternen reflektiert als auch eigenes Licht emittiert (Ionisation).

Bei dem Teleskop handelt es sich um das eVscope 2.0 von Unistellar. Laut der Metadaten ist ein IMX347 CMOS-Sensor von Sony verbaut, der eine Auflösung von 4,17 Megapixeln haben soll. Die Aufnahmen selbst sind 2088x1536 px groß und haben eine Farbtiefe von 12 Bit, die im tiff-Format als 16 Bit gespeichert werden. Es handelt sich um 108 Bilder. Die Belichtungszeit liegt bei etwa 4 Sekunden pro Bild.

3 Pre-Processing

Als erster Schritt müssen die einzelnen Aufnahmen geladen und vorverarbeitet werden. In Matlab werden die Bilder als eine uint16-Matrix in der Größe der Auflösung dargestellt, wobei jeder Eintrag den Grauwert eines Pixels repräsentiert. Um systematische Fehler beim Auslesen des Sensors zu korrigieren, wurde zusätzlich zu den Rohaufnahmen eine Aufnahme ohne Beleuchtung erstellt, die als Dunkelbild dient. Hier erkennt man einige *Hotpixel*, die in jeder der Aufnahmen vorhanden sind, jedoch sind nicht alle im Dunkelbild enthalten, da dieses unter anderen Bedingungen aufgenommen wurde. Um die restlichen Hotpixel zu finden, werden Pixel die in mindestens 90 % der Bilder deutlich heller als ihre Nachbarn sind identifiziert. Diese werden dann in einer Hotpixelmaske gespeichert. Dann werden sowohl Hotpixel als auch das Dunkelbild von den Rohaufnahmen subtrahiert. Die Pixel die einen negativen Wert haben, werden auf 0 gesetzt.

3.1 Rekonstruktion der Farben

Bis zu diesem Schritt wurden die Aufnahmen als Graustufenbilder verarbeitet. Der vom Teleskop verwendete Sensor ist jedoch ein Farb-CMOS-Sensor, der die Farben mit Hilfe eines Bayer-Filters aufnimmt. Dabei sind die Pixel des Sensors in einem Schachbrettmuster mit den Farben Rot, Grün und Blau angeordnet. In den Meta-Daten der Aufnahmen ist die Reihenfolge dieser Anordnung mit *rggb* angegeben. Mit der demosaicing-Funktion in Matlab können die Farbinformationen aus den Graustufenbildern interpretiert und mittels Interpolation rekonstruiert werden.

```
img_rgb = demosaic(img_calibrated, 'rggb', BitsPerSample=16);
```

Dabei wird die Matrix, die das Bild darstellt auf drei Dimensionen erweitert, um die Farbkanäle Rot, Grün und Blau zu repräsentieren. Die Aufspaltung in die Farbkanäle muss vor der Ausrichtung der Bilder erfolgen, da ansonsten die Interpretation der Pixelfarbe ohne weitere Maßnahmen durcheinander kommt.

Bevor die Verarbeitung weitergeht, werden die Bilder noch grob zugeschnitten, um irrelevante Bildbereiche zu entfernen, die die Identifizierung von Sternen stören könnten.

Es kann auch hilfreich sein, nicht alle Bilder für die weitere Verarbeitung zu verwenden. In den Meta-Daten der Aufnahmen kann man erkennen, dass das Teleskop eigene Verarbeitungsverfahren 37 der Rohaufnahmen aufgrund von schlechter Qualität ausgeschlossen hat.

4 Ausrichtung und Stacking

Nachdem die Bilder vorverarbeitet wurden, können sie ausgerichtet und gestapelt werden. Da die Aufnahmen nacheinander gemacht wurden, haben sich die Sterne auf den Bildern aufgrund der Erdrotation verschoben. Kann man ihre Position bestimmen, kann man die Bilder entsprechend ausrichten.

4.1 Stern-Erkennung

Für die Sternerkennung wird nur die Helligkeit der Pixel betrachtet. Mit einem Schwellwert wird das Hintergrundrauschen reduziert:

```
I_thresh = 0.6;
I_cutoff(I_cutoff < I_thresh) = 0;
```

Dann wird das Bild mit einem Gauß-Filter geglättet, was das Rauschen etwas reduziert und das Leuchten von Sternen weichzeichnet:

```
I_smooth = imgaussfilt(I_cutoff, 2);
```

Nun werden die Pixel maskiert, die heller als ihre Nachbarn sind (lokale Maxima):

```
bwMax = imregionalmax(I_smooth);
```

Für jedes dieser Pixel wird der Helligkeits-Schwerpunkt bestimmt, also das Zentrum des Sterns. Das ist sinnvoll, um bei größeren Objekten eine genauere Position zu erhalten:

```
stats = regionprops(bwMax, I, 'WeightedCentroid');
```

Die Sterne werden dann noch nach ihrer Helligkeit sortiert und die Position der 10 hellsten als Referenz gespeichert. Abbildung 1 zeigt die Positionen dieser Sterne im Bild.



Abbildung 1: Positionen der 10 hellsten Sterne im Bild in Rot

4.2 Transformation

Bevor versucht werden kann die Positionen der Sterne mit denen aus einem Referenzbild abzugleichen, müssen die Positionen bestimmt werden, die in beiden Bildern vorhanden sind. Das liegt daran, dass aufgrund von Variationen in der Helligkeit oder der Bildqualität nicht zwangsläufig dieselben Sterne in beiden Bildern erkannt werden. Da die Bilder nur leicht verschoben sind, kann einfach über die Distanz der Positionen ein *Nearest-Neighbour*-Abgleich erfolgen.

Sind nun die Positionen derselben Sterne im Referenzbild als auch in dem aktuellen Bild bestimmt, kann eine Transformation berechnet werden, die die Positionen im aktuellen Bild an die Positionen im Referenzbild anpasst:

```
tform = estimateGeometricTransform2D(i_points, r_points, 'affine');
```

Mit dieser Methode wird eine *affine* Transformation bestimmt, das bedeutet Translation, Rotation, Skalierung und Scherung, werden berücksichtigt. Die errechnete Transformationsmatrix kann nun auch auf das Bild angewendet werden:

```
aligned = imwarp(moving_image, tform, "OutputView", output_view);
```

Der Unterschied zwischen den Positionen der Sterne und ihre Transformation werden in Abbildung 2 dargestellt.

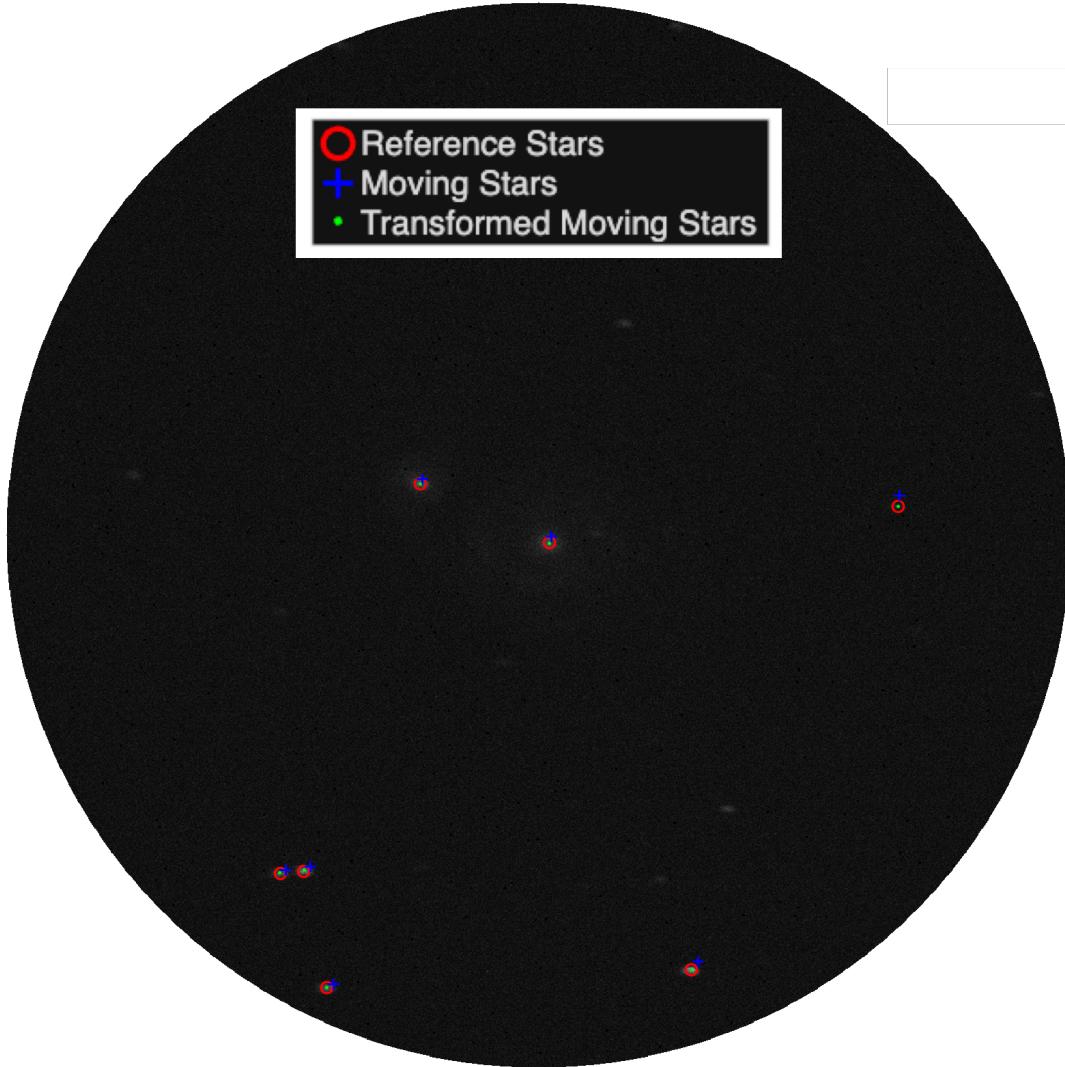


Abbildung 2: Sternpositionen in Referenz- und aktuellem Bild (Reference/ Moving), die Positionen der Referenzsterne in Rot, die Positionen der aktuellen Sterne in Blau und die transformierten Positionen in Grün.

4.3 Stacking

Nach der Ausrichtung sind die Bilder bereit *gestapelt* zu werden. Das bedeutet, dass die Bilder summiert werden und ein Mittelwert gebildet wird. Dies geschieht pixelweise für jeden Farbkanal. Außerdem kann zusätzliche Rauschunterdrückung über eine Filterung der Standardabweichung erreicht werden. Dabei wird über alle Bilder hinweg der Mittelwert eines Pixels und die Standardabweichung berechnet. Anschließend werden Pixelwerte, die zu stark vom Mittelwert abweichen mit einer Maske ignoriert:

```

pixel_mean = mean(stack, 4);
pixel_std  = std(stack, 0, 4);
k = .12;
mask = abs(stack - pixel_mean) <= k*pixel_std;

```

Das wird als sogenanntes σ -Clipping bezeichnet.

Durch das Stapeln der Bilder wird das Signal-zu-Rauschverhältnis stark verbessert, dennoch ist das resultierende Bild noch fast völlig schwarz. Weitere Verarbeitung ist nötig.

5 Post-Processing

Um die Whirlpool-Galaxie sichtbar zu machen, muss der Kontrast und der Helligkeitsbereich des Bilds angepasst werden. Das Teleskop gibt zusätzlich zu den Rohaufnahmen noch einen Fit der Hintergrundbeleuchtung aus. Dieser kann verwendet werden, um störendes Hintergrundlicht abzuziehen. Dafür muss er auch zunächst in RGB umgewandelt werden. Außerdem kann mit einem konstanten Faktor pro Farbkanal ein konstanter Offset hinzugefügt werden, um den Hintergrund weiter zu reduzieren und den *Schwarzpunkt* anzupassen. Alle negativen Werte werden auf 0 gesetzt.

Es folgt ein Schritt der Rauschunterdrückung. Dafür wird ein bilateraler Filter angewendet, der Anhand der Standardabweichung von Pixeln zu ihrer Nachbarschaft arbeitet.

Um den Kontrast anzupassen wird zunächst eine graue Maske erstellt, die den Bildinhalt von dem Hintergrund unterscheidet. Sie ist in Abbildung 3 dargestellt. Aus ihr wird ein Faktor errechnet, mit dem das Bild skaliert wird:

```

scale_factor = 1 + mask * 2;
img_denoised = img_denoised ./ scale_factor .* galaxy;

```

Es wird eine weitere Maske für die Sterne errechnet, indem die Maske mit sich selbst multipliziert und geglättet wird. Diese kann dann von der Hintergrundmaske subtrahiert werden:

```

stars = mask.^1.5;
stars = imgaussfilt(stars, 5);
galaxy = (mask - stars * 1.1);

```

So kann eine Maske der Galaxie erstellt werden, die benutzt wird, um die Helligkeit besser anpassen zu können.

Dann wird das Bild noch mit einer Gamma-Korrektur skaliert, um die Helligkeitsverteilung anzupassen. Dafür wird Matlab's *imadjust* Funktion für jeden Farbkanal verwendet:

```

low_in = prctile(ch(:, 50);
high_in = prctile(ch(:, 99.99);
gamma = .7;
img_adj(:, :, k) = imadjust(img(:, :, k), [low_in, high_in], [0, 1], gamma);

```

Dabei handelt es sich um eine nicht-lineare Anpassung, die schwächere Helligkeitswerte stärker betont. Danach kann sich die Galaxie besser vom Hintergrund abheben.

Schließlich werden die Farbkanäle noch einzeln gewichtet, um ein schöneres Gesamtbild zu



Abbildung 3: Graue Maske zur Unterscheidung von Bildinhalt und Hintergrund

erzeugen. Dazu werden die Gewichtungen für jeden Farbkanal manuell festgelegt. Im resultierenden Bild, dargestellt in Abbildung 4, ist die Galaxie deutlich zu erkennen.

6 Auswertung

Das Ergebnis zeigt, dass die Verarbeitung trotz der schwachen Lichtintensität der Galaxie in den Aufnahmen ein deutliches Bild der Whirlpool-Galaxie produzieren kann. Das bedeutet, dass das Ausrichten der Bilder erfolgreich implementiert wurde und eine Nachbearbeitung anhand sinnvoller Werte durchgeführt wurde. Besonders gelungen ist die effektive Entfernung aller Hotpixel.

Als Vergleich zu dem Bild, was im Rahmen dieser Ausarbeitung erstellt wurde, kann das von der Software des Teleskops automatisch erzeugte Bild herangezogen werden. Dieses Bild ist in Abbildung 5 dargestellt. Bei genauerer Betrachtung ist es deutlich zu erkennen, dass trotz aller bereits beschriebenen Verarbeitungsschritte noch ein großer Unterschied zwischen dem in Matlab bearbeiteten und dem automatisch erzeugten Bild besteht. Besonders die Schärfe von Details im Nebel und den Sternen ist in letzterem deutlich ausgeprägter.



Abbildung 4: Das finale Bild der Whirlpool-Galaxie nach der Verarbeitung

Wahrscheinlich sind bereits bei der Rohdatenverarbeitung die Bilder anhand ihrer Qualität selektiert worden. Vor allem aber scheinen die Rauschunterdrückungsalgorithmen in der Nachbearbeitung des Teleskops sehr effektiv zu sein. Während dieser Ausarbeitung wurden verschiedene Ansätze ausprobiert, um das Rauschen in der Nachbearbeitung zu reduzieren. Dabei fällt auf, dass die meisten Methoden sehr sensitiv gegenüber der Eingangsparameter sind. Eine Filterung, die konsistente Ergebnisse für unterschiedliche Aufnahmen liefert zu finden, ist schwierig. Aktuelle Ansätze in der astronomischen Bildverarbeitung verwenden teilweise Machine Learning [2].

Zusammenfassend kann man sagen, dass ein anschauliches Bild der Whirlpool-Galaxie im Rahmen dieser Ausarbeitung erzeugt werden konnte, auch wenn es noch viel Verbesserungspotential im Vergleich der Verarbeitungsqualität zur automatisch erstellten Aufnahme gibt.

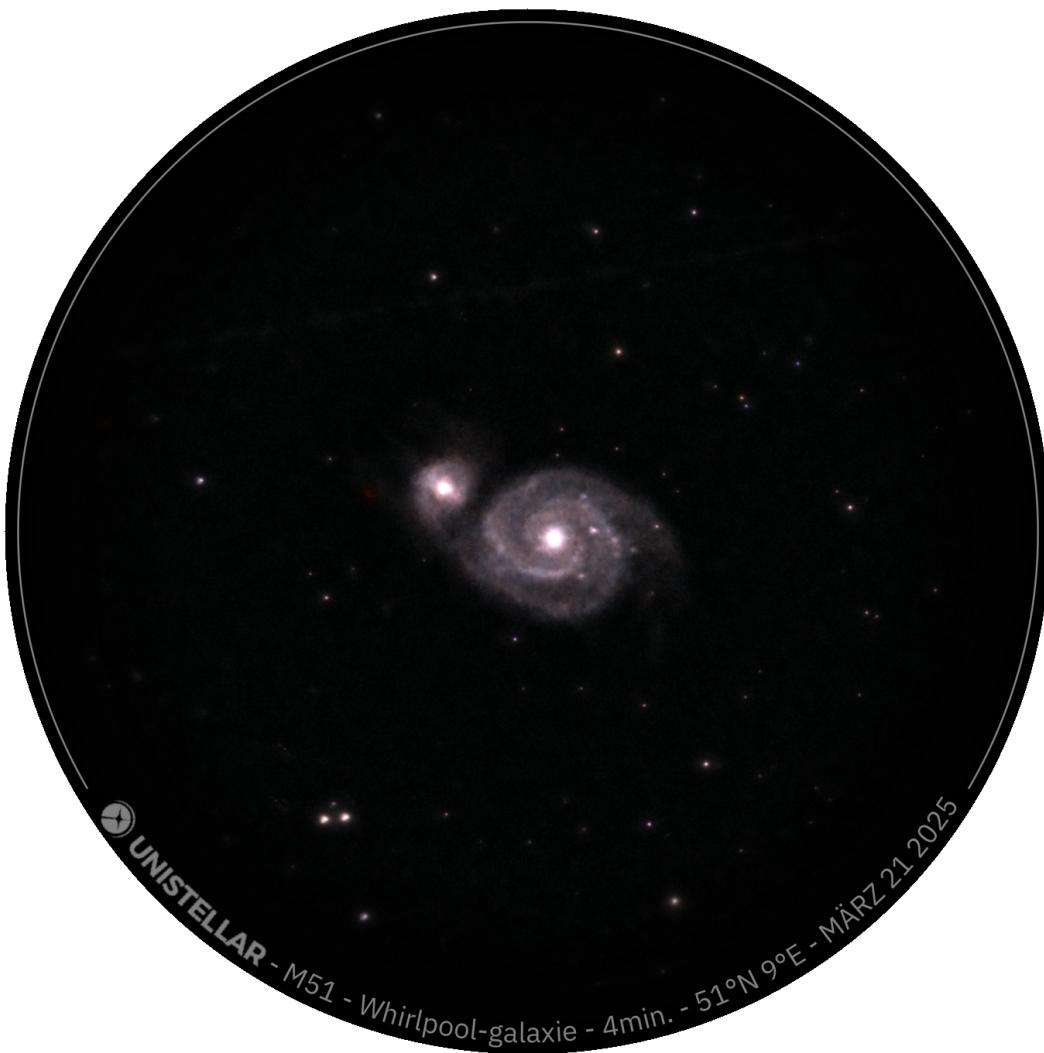


Abbildung 5: Das vom Teleskop errechnete Bild der Whirlpool-Galaxie

7 Quellcode

Der gesamte Quellcode wird zusammen mit diesem Dokument abgegeben und ist auf <https://github.com/LoloSpirit/galaxy-image-enhancement> frei verfügbar. Das Live-Script WORKFLOW.MLX im CODE-Ordner enthält den gesamten Verarbeitungsworkflow. Einzelne Funktionen sind als separate Dateien im selben Ordner hinterlegt. Der Code und alle Kommentare sind in englischer Sprache verfasst.

Literatur

- [1] Rafael C. Gonzalez und Richard E. Woods. *Digital Image Processing*. London, UK: Pearson, 2018.
- [2] Edward Higson. “Reconstructing Astronomical Images with Machine Learning”. In: *Physical Sciences* (2019).