

BlurXTerminator

Version 1.0.2 AI

Von Russell Croman, RC Astro, LLC

Deutsche Übersetzung von W.M. Kasakow mit Erlaubnis von Russel Croman

Deconvolution astronomischer Bilder mit KI-Methoden (neuronale Netze).

Inhalt

1 Beschreibung

2 Anwendung

2.1 Anpassungen

- 2.1.1 Stellare Anpassungen
- 2.1.2 Nichtstellare Anpassungen
- 2.1.3 Optionen

2.2 Eingaben

- 2.2.1 Lineare Daten
- 2.2.2 Sterne
- 2.2.3 Farbe
- 2.2.4 Rauschen
- 2.2.5 Sampling
- 2.2.6 Bessere Daten
- 2.2.7 Bilder des Weltraumteleskops
- 2.2.8 Bilder mit hohem Dynamikbereich unter MacOS

3 Erörterung

- 3.1 Wie funktioniert BlurXTerminator?
- 3.2 Dekonvolution? Wirklich?
- 3.3 Perfektion?
- 3.4 Vermeiden Sie eine Überverarbeitung
- 3.5 Zukünftige Verbesserung
- 3.6 Support
- 3.7 Eine technische Analyse von Auflösung und Sampling

- 3.7.1 Analyse von Auflösung und Sampling unter Verwendung von Signalverarbeitungskonzepten

3.8 GPU-Beschleunigung

- 3.8.1 Kompatibilität
- 3.8.2 Laden Sie das NVIDIA CUDA-Toolkit herunter und installieren Sie es
- 3.8.3 CuDNN-Dateien herunterladen und installieren

- 3.8.4 Laden Sie die ZLIB-Komprimierungsbibliothek herunter und installieren Sie sie
- 3.8.5 Laden Sie die GPU-fähige TensorFlow-Bibliothek herunter und installieren Sie sie
- 3.8.6 Verifizieren/Setzen von Umgebungsvariablen
- 3.8.7 Genießen Sie die schnelle neuronale Netzwerkverarbeitung

1. Beschreibung

BlurXTerminator ist ein KI-basiertes Dekonvolution-Tool, das speziell für astronomische Bilder entwickelt wurde, die mit Equipment aufgenommen wurden, das üblicherweise von Amateur-Astrofotografen verwendet wird. Nicht alle KI sind gleich. Es gibt KI-basierte Schärfungswerkzeuge für die allgemeine Fotografie, aber wenn sie auf astronomische Bilder angewendet werden, neigen sie dazu, Details zu „erfinden“, die es nicht gibt. Sie gehen normalerweise auch nicht sehr gut mit Sternen um. Ihre neuronalen Netzwerke wurden nicht auf astronomische Bilder trainiert, daher machen sie oft schlechte „Vermutungen“, wie die ursprüngliche, unverschommene Szene aussieht.

Die Designabsicht von BlurXTerminator ist es, basierend auf tatsächlich in einem Bild vorhandenen, kontrastarmen Informationen so viele Details wie möglich wiederherzustellen, ohne Details zu fabrizieren, die tatsächlich nicht vorhanden sind, nur um ein Bild zu erhalten, das schärfer **erscheint**. Bei der Architektur und dem Training des neuronalen Netzwerks wurde große Sorgfalt darauf verwendet, sicherzustellen, dass seine Ergebnisse bei richtiger Verwendung so real wie möglich sind.

Jede Dekonvolution, einschließlich der klassischen Algorithmen, die von Richardson, Lucy, van Cittert und anderen entwickelt wurden, beinhaltet im Wesentlichen Vermutungen. Mathematisch wird die Dekonvolution als ein *schlecht gestelltes* Problem bezeichnet: Für ein gegebenes, verschwommenes Eingabebild gibt es viele mögliche schärfere Bilder, die bei erneuter Unschärfe zu demselben Eingabebild führen würden. Welche ist richtig oder zumindest eine bessere Vermutung?

Die klassischen Algorithmen verwenden die Kenntnis der *Point-Spread-Funktion* (PSF oder Punktverteilungs-Funktion) eines Bildes, um die Dekonvolution zu unterstützen, die allerdings nur funktioniert, solange die dem Algorithmus zugeführte PSF relativ genau ist. Die Anwendung neuronaler Netze zur Dekonvolution bringt eine zusätzliche Informationsquelle zur Steuerung des Prozesses: Kenntnis der Strukturen und Muster, die typischerweise in echten, hochauflösenden astronomischen Bildern vorhanden sind. Das neuronale Netzwerk von BlurXTerminator wurde mit extrem hochauflösenden Bildern trainiert, die von Teleskopen wie dem Hubble oder James Webb aufgenommen wurden. Es „versteh“t, wie astronomische Strukturen tatsächlich in feineren Maßstäben aussehen, als sie mit Amateurgeräten aufgelöst werden können.

Die Trainingsmethodik umfasst zusätzlich ein tiefes Verständnis der gemeinsamen Point-Spread-Functions, denen astronomische Bilder unterliegen, einschließlich Variationen, die durch atmosphärische Turbulenzen, optische Streuung, Erfassungsprobleme wie Guidingfehler und optische Verzerrungen wie Koma und chromatische Aberration verursacht werden. Es ist nicht erforderlich, das PSF im Voraus zu extrahieren: BlurXTerminator verwendet die Sterne in einem Bild als PSF-Referenzen. Es analysiert und verarbeitet ein Bild in einem Schritt, wobei in den meisten Fällen *keine* Iteration erforderlich ist.

BlurXTerminator kann unterschiedliche Mengen an Dekonvolution auf die stellaren und nicht-stellaren Bereiche eines Bildes anwenden. Der Versuch, alle verfügbaren Details in nichtstellaren, ausgedehnten Objekten mit den klassischen Algorithmen wiederherzustellen, führt normalerweise zu dunklen Halos (Ringing) um Sterne. Mit BlurXTerminator kann mehr Schärfe auf die nicht-stellaren Teile eines Bildes angewendet werden, wodurch mehr Details herausgeholt werden, ohne dass (in den meisten Fällen) Ringartefakte um die verkleinerten Sterne erzeugt werden.

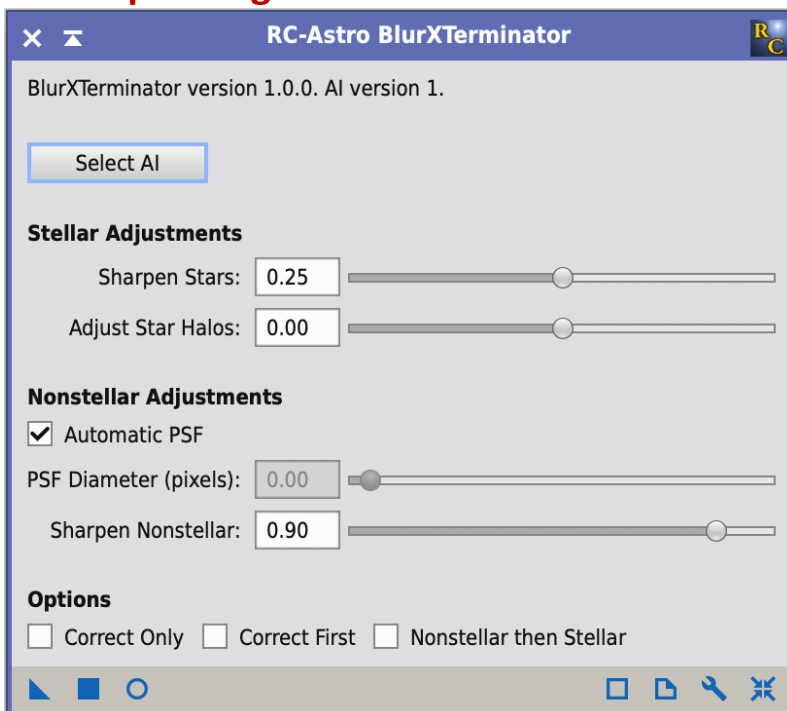
BlurXTerminator kann zusätzlich andere in einem Bild vorhandene Aberrationen in begrenztem Umfang korrigieren. Derzeit sind dies für die meisten astronomischen Instrumente:

- Guidingfehler
- Astigmatismus
- Primäres und sekundäres Koma
- Chromatische Aberration (Farbsäume)
- Unterschiedliche Sterndurchmesser (FWHM) und Lichthöfe in jedem Farbkanal

Es wird nicht davon ausgegangen, dass diese Aberrationen stationär sind: Sie können über das Sichtfeld variieren. Dies ist ein großer Vorteil gegenüber klassischen Dekonvolutionsalgorithmen, die davon ausgehen, dass dieselbe PSF für das gesamte Bild gilt. Beispielsweise werden Sterne mit begrenztem Komaprofil in den Ecken eines Bildes abgerundet und dann geschärft, während Sterne in der Bildmitte, die bereits rund sind, einfach geschärft werden. Diese Korrektur kann auch auf die nichtstellaren Merkmale des Bildes angewendet werden. Die Korrektur kann als separater Schritt oder in Kombination mit dem Schärfen erfolgen.

2. Anwendung

2.1 Anpassungen



2.1.1 Stellare Anpassungen (Stellar Adjustments)

Sterne schärfen (Sharpen Stars)

Betrag zur Verringerung des Durchmessers (im Sinne von FWHM) von Sternen. Höhere Werte führen zu kleineren, schärferen Sternen, die nur halb so groß sind wie ihr ursprünglicher Durchmesser.

Hohe Werte für diese Einstellung funktionieren möglicherweise nicht gut für alle Bilder, die von allen Instrumenten erfasst werden. Wählen Sie eine Einstellung, die zu einer

angemessenen Reduzierung des Sterndurchmessers führt, ohne Artefakte zu erzeugen. Wenn dunkle Halos um Sterne herum erscheinen (Ringing), wählen Sie einen niedrigeren Wert oder kompensieren Sie dies, indem Sie den Parameter „Adjust Star Halos“ erhöhen.

Bei Bildern, die mit Instrumenten mit langer Brennweite aufgenommen wurden, können hohe Werte dieses Parameters auffällige strukturlose Bereiche um hellere Sterne hinterlassen: Es gibt wenig bis gar keine Informationen im Bild, mit denen diese Pixel mit Details gefüllt werden könnten, da durch Übersättigung die Mitten der Sternprofile abgeschnitten werden.

Sternhalos anpassen (Adjust Star Halos)

Einstellung zum Anpassen der "Halos" von Sternen. Höhere Werte führen zu helleren Halos mit größerer Ausdehnung, was den Sternen ein weiches Aussehen verleiht. Kleinere Werte verringern die Halo-Helligkeit und -Ausdehnung und verleihen den Sternen ein schärferes, härteres Aussehen.

Niedrige Werte für diese Einstellung können in Bildern, die mit einigen Instrumenten aufgenommen wurden, dunkle Halos um Sterne hinterlassen. Wählen Sie eine Einstellung, die zu einer angemessenen Anpassung der Halos ohne Artefakte führt.

Da die Farben von Sternen mit gesättigten (geclippten) Kernen nur in ihren Halos sichtbar sind, kann eine Vergrößerung ihrer Halos bei gleichzeitiger Verringerung des Sterndurchmessers dazu beitragen, die Sternfarbe beizubehalten oder sogar noch hervorzuheben.

2.1.2 Nichtstellare Anpassungen (Nonstellar Adjustments)

Automatisches PSF (Automatic PSF)

Ermittelt die PSF (im FWHM-Sinne) für nichtstellares Schärfen automatisch aus dem Bild.

Damit die PSF genau genug wird, müssen in allen Teilen des Bildes Sterne vorhanden sein. Wenn keine Sterne vorhanden sind, versucht BlurXTerminator, die PSF adaptiv aus nicht-stellaren Bildmerkmalen zu bestimmen. Dies kann in qualitativer Hinsicht bei vielen Bildern immer noch gut funktionieren, aber schwächere Strukturen, die zufällig auch in Wirklichkeit eher ein verschwommenes Aussehen haben, können damit überschärft werden.

Wenn „Automatisches PSF“ ausgewählt ist, wird jede Aberrationskorrektur (Koma, Astigmatismus usw.), die auf Sterne angewendet wird, auch auf nichtstellare Merkmale angewendet. Dies gilt nicht im manuellen Modus (es sei denn, die Option „Correct First“ ist ausgewählt): Die PSF von nichtstellaren Merkmalen wird als rund angenommen.

Einige Bilder, insbesondere solche, die mit langen Brennweiten aufgenommen wurden, können Bereiche mit wenigen oder keinen Sternen aufweisen. BlurXTerminator kann die nicht-stellaren Merkmale in diesen Bereichen überschärfen, wenn die automatische PSF-Option ausgewählt ist. Bilder werden in „Kacheln“ mit 512 x 512 Pixeln verarbeitet, wobei sich die Kacheln um 20 % überlappen, um Artefakte zu vermeiden. Einzelne Kacheln werden zu einem bestimmten Grad unabhängig verarbeitet, um nicht-stationäre PSFs zu ermöglichen. Wenn eine bestimmte Kachel nicht genügend Sterne enthält, kehrt BlurXTerminator zu dem Versuch zurück, die PSF von nicht-stellaren Merkmalen abzuleiten, was zu einer ungleichmäßigen Schärfung im Bild führt.

In diesen Fällen erhalten Sie genauere Ergebnisse, indem Sie in den manuellen Modus wechseln und den PSF-Durchmesser auf die bekannte FWHM der Sterne im Bild als vernünftigen Ausgangspunkt einstellen.

PSF-Durchmesser (PSF Diameter (pixels))

Hier kann man den PSF-Durchmesser (FWHM) eintragen, der bei der Deconvolution *nichtstellarer* Merkmale verwendet werden soll.

Für die beste Genauigkeit sollte dies auf die FWHM von Sternen im Bild oder für Mond- und Planetenbilder auf die FWHM von Sternen eingestellt werden, die von demselben optischen und Kamerasystem unter Verwendung derselben Vorverarbeitungsmethodik erzeugt wurden.

Bilder mit FWHM-Werten über 8 Pixel, dem Höchstwert für diesen Parameter, sind wahrscheinlich um mehr als den Faktor zwei oversampled und können sicher um den Faktor zwei heruntergesampelt (Downsampling) werden. Siehe Punkt „2.2.5 Sampling“ für weitere Informationen.

Nichtstellar schärfen (Sharpen Nonstellar)

Hier kann der Betrag eingetragen werden, wie stark nichtstellare Bildmerkmale zu schärfen sind.

BlurXTerminator versucht, die Größe der PSF von nicht-stellaren Bildmerkmalen bis zum angegebenen vollen Durchmesser zu reduzieren. Mit anderen Worten, die Einstellung auf 1,00 bedeutet, dass versucht wird, die nichtstellare PSF auf die Größe Null zu reduzieren – eine ideale Punkt-PSF und damit die maximal mögliche Menge an Schärfe. Die *tatsächlich* erzielte Schärfung wird durch die wiederherstellbaren, kontrastarmen Details begrenzt, die im Bild bei feinen Pixelskalen vorhanden sind.

Der Wert 0.00 bedeutet keine Schärfung und 1.00 größtmögliche Schärfung.

2.1.3 Optionen (Options)

Nur korrigieren (Corret only)

BlurXTerminator kann begrenzte Mengen an Bewegungsunschärfe (Guidingfehler), Astigmatismus, primäre und sekundäre Koma, ungleiche FWHM in Farbkanälen, leichte chromatische Aberration und asymmetrische Sternhalos korrigieren. Die Korrektur wird immer durchgeführt, aber manchmal können bessere Ergebnisse erzielt werden, indem die Korrektur als separater Schritt *vor* jedem weiteren Schärfen durchgeführt wird.

Dies ist eine reine Komfortoption, die der Auswahl von „Automatische PSF“ und dem Setzen aller anderen Parameter auf Null entspricht. Die Korrektur wird sowohl auf nichtstellare Merkmale als auch auf Sterne angewendet, solange Sterne in allen Teilen des Bildes vorhanden sind.

Technisch gesehen versucht BlurXTerminator, die im Bild gefundene Point-Spread-Funktion (PSF) *azimutal symmetrisch* (rund) zu machen. Die PSF muss nicht *stationär* sein – die Aberrationen können über das Bild hinweg variieren. BlurXTerminator versucht, die lokale PSF in jedem Teil des Bildes zu korrigieren, wodurch die Sterne rund werden, während ihre Schwerpunkte erhalten bleiben.

Zuerst korrigieren (Correct first)

Dies ist eine praktische Option, bei der das Bild zweimal verarbeitet wird: einmal wie bei der Option „Nur korrigieren“ oben und dann mit anderen aktivierten Anpassungen. Dies wird schneller ausgeführt, als jeden Schritt separat auszuführen, da der Verarbeitungsaufwand zwischen beiden Operationen geteilt wird.

Nichtstellar dann Stellar (Nonstellar then Stellar)

Führt zuerst nicht-stellares Schärfen durch, gefolgt von stellarem Schärfen.

Dies kann für einige Bilder nützlich sein, zum Beispiel Galaxien mit schwachen, kaum aufgelösten Sternen, die in andere Strukturen wie HII-Regionen eingebettet sind. Das Schärfen von Sternen erkennt diese möglicherweise zunächst nicht als Sterne, aber wenn vor dem Schärfen von Sternen ein nicht-stellares Schärfen durchgeführt wird, werden diese Sterne im ersten Schritt aufgelöst und dann im zweiten Schritt geschärft. Hier ist probieren angesagt.

2.2 Eingaben

2.2.1 Lineare Daten

Wenn Sie ein neues Tool zum ersten Mal verwenden, ist es verlockend, es an bereits bearbeiteten Bildern auszuprobieren. BlurXTerminator funktioniert wie klassische Dekonvolutionsmethoden am besten mit *linearen* Bilddaten, idealerweise direkt nach der Integration, der Kanalkombination und vielleicht der Farbkalibrierung und der Hintergrundglättung, aber **vor** jeder weiteren Verarbeitung. Die einzige Ausnahme ist, dass ein *angemessenes* Maß an einfacher Streckung mit dem Histogramm-Transformation-Tool die Leistung nicht beeinträchtigt. Jede andere Streckungsmethode und überhaupt jede andere Verarbeitung führt wahrscheinlich zu einer weniger genauen Leistung und deshalb zwangsläufig zu Artefakten: Diese Schritte bitte erst nach der Deconvolution mit BlurXTerminator ausführen.

Es wird beispielsweise nicht empfohlen, vor der Verwendung von BlurXTerminator eine Rauschunterdrückung durchzuführen. Das neuronale Netzwerk von BlurXTerminator wurde darauf trainiert, Details in auch Gegenwart von Rauschen wiederherzustellen. Die meisten Rauschunterdrückungstechniken verändern oder zerstören die dafür benötigten, kontrastarmen Informationen auf feinen Pixelskalen.

Die Anwendung klassischer Dekonvolutionsalgorithmen vor oder nach der Anwendung von BlurXTerminator wird ebenfalls nicht empfohlen. Das resultierende Bild kann dann vielleicht sogar schärfer *erscheinen*, aber die Wahrscheinlichkeit, dass dieses scheinbare zusätzliche Detail nicht der Realität entspricht, wird deutlich erhöht. Außerdem führt die Anwendung von BlurXTerminator *nach* einer zuerst durchgeführten, herkömmlichen Dekonvolution normalerweise dazu, dass leichtes kräuseln oder wurmartige Strukturen, die manchmal durch herkömmliche Algorithmen erzeugt werden, deutlich verstärkt werden.

Bilddaten sollten im Fließkommaformat vorliegen, auch wenn sie mit dem Histogramm-Transformations-Tool gestreckt sind. Wenn ein gestrecktes Bild vor der Anwendung von BlurXTerminator in ein 16-Bit-Ganzzahlformat konvertiert wird, können sehr helle Bereiche eine Posterisation (Tonwerttrennung) entwickeln, eine schrittweise Quantisierung der Helligkeit und/oder der Farbe im Bild.

2.2.2 Sterne

BlurXTerminator verwendet die Sterne in einem Bild als Referenz, um zu verstehen, wie es unscharf wurde und wie dies über das Sichtfeld hinweg variiert. Wenn Sie die Sterne aus dem Bild entfernen möchten, führen Sie diesen Vorgang am besten **nach** der Anwendung von BlurXTerminator durch.

Wenn die Sterne vor der Anwendung von BlurXTerminator trotzdem entfernt werden, wird empfohlen, die Option „Automatische PSF“ zu deaktivieren und stattdessen den PSF-Durchmesser für nicht-stellares Schärfen manuell auf den FWHM-Wert der Sterne vor deren Entfernung einzustellen. Wenn der automatische Modus ausgewählt ist, versucht BlurXTerminator adaptiv, die lokale PSF aus nicht-stellaren Merkmalen im Bild zu bestimmen. Während dies zu einem Ergebnis führen kann, das im qualitativen Sinne „gut aussieht“, führt es normalerweise zu einer Überschärfung kontrastarmer Bereiche, denen es in Wirklichkeit an Details eher mangelt.

Wenn es Teile eines Bildes gibt, die wenige oder keine Sterne innerhalb eines Messbereichs (einzelne Kachel) haben, kann dasselbe passieren: Überschärfen von kontrastarmen Bereichen. Verwenden Sie in diesem Fall den manuellen PSF-Durchmesserparameter (PSF Diameter) und stellen Sie ihn auf den FWHM-Wert der Sterne im Bild als Ausgangspunkt ein. Dadurch wird sichergestellt, dass das gesamte Bild gleichmäßig geschärft wird.

Für Bilder ohne Sterne, wie z. B. Mond- und Planetenbilder, wird der manuelle PSF-Durchmessermodus (PSF-Diameter) für beste Genauigkeit empfohlen. Stellen Sie ihn auf den FWHM-Wert von Sternen ein, den dasselbe Bildgebungssystem und dieselbe Vorverarbeitungslinie erzeugen würden, zumindest als Ausgangspunkt.

2.2.3 Farbe

Die besten Ergebnisse werden höchstwahrscheinlich erzielt, wenn BlurXTerminator auf RGB-Bildern ausgeführt wird, im Gegensatz zu jedem monochromen Kanal unabhängig voneinander. Das neuronale Netz wird darauf trainiert, nicht nur die PSF in jedem Kanal zu erkennen, sondern auch die Beziehungen zwischen den einzelnen Kanälen. Leichte Farbsäume, die beispielsweise durch chromatische Aberration entstehen, sind in den Daten eines einzelnen Kanals nicht sichtbar und können von BlurXTerminator nicht korrigiert werden, es sei denn, es sieht alle drei Kanäle gleichzeitig und kann dies berücksichtigen.

Beim Kombinieren von Schmalbanddaten mit einem Farbbild ist es am besten, vor der Dekonvolution das „Mischen“ von Farbkanälen zu vermeiden, wie es bei einigen fortgeschrittenen Farbpalettentechniken der Fall ist. Wenn Kanäle vor der Deconvolution gemischt werden, insbesondere mit sehr unterschiedlichem Gain, können Sternprofile erheblich verändert werden, vielleicht so sehr, dass sie für das neuronale Netzwerk nicht mehr als Sterne erkennbar sind. Führen Sie besser eine einfache SHO-Farbkombination durch, dann BlurXTerminator und erst anschließend eine Mischung zwischen den Schmalband und den RGB-Kanälen durch.

Aus ähnlichen Gründen kann es bei der Verarbeitung von LRGB-Bildern am besten sein, die L- und RGB-Bilder *separat* einer Dekonvolution zu unterziehen und dann erst zu einem LRGB zu kombinieren. Wenn sich Sternprofile in L erheblich von denen in RGB unterscheiden, kann nämlich die LRGB-Kombination zu Sternprofilen führen, die das neuronale Netzwerk nur schwer erkennen kann.

2.2.4 Rauschen

Aus Bildern mit höherem SNR können mehr Details wiederhergestellt werden. Fertigen Sie also mehr Rohbilder an und integrieren Sie sie, um das SNR zu erhöhen.

Eine grundsätzliche Grenze für jede Informationswiederherstellungstechnik wie Dekonvolution, ist die Menge des vorhandenen Signals im Vergleich zur Menge des Rauschens, das sogenannte *Signal-Rausch-Verhältnis* (SNR). Weder BlurXTerminator noch andere Verarbeitungswerkzeuge oder -Techniken können diese Einschränkung überwinden, die ein Grundprinzip der Informationstheorie ist. In einer photonenbegrenzten Praxis wie der Astrofotografie besteht die **einzig**e Möglichkeit, das SNR zu erhöhen, bei sonst gleichen Bedingungen darin, mehr Licht zu sammeln, das heißt mehr Belichtungen auszuführen. Dies setzt voraus, dass das Rauschen bei Einzelaufnahmen tatsächlich durch Schrotrauschen begrenzt wird, das Rauschen, das der Messung eines Signals in Form diskreter Ereignisse (in diesem Fall das Eintreffen einzelner Photonen) innewohnt.

2.2.5 Sampling

Sie werden feststellen, dass der maximale PSF-Durchmesser-Parameterwert 8 Pixel beträgt. Dies ist der maximale PSF-Durchmesser (im Sinne von FWHM), für dessen Entfaltung BlurXTerminator entwickelt wurde. Was ist mit Bildern, die größere Sterne und damit größere Unschärfen haben?

Es ist mathematisch einfach zu demonstrieren, dass Bilder mit PSF-FWHM-Werten von mehr als 8 Pixeln mindestens um den Faktor zwei oversampled sind: Sie enthalten keine signifikanten Informationen bei Skalen, die feiner als zwei Pixel sind. Diese Bilder können daher sicher (z. B. mit dem IntegerResample-Prozess) ohne erkennbaren Informationsverlust um den Faktor zwei heruntergesampelt (Downsampling) werden. Das SNR wird aufgrund des durchschnittlichen Rauschens über mehrere Pixel höher sein, die nachfolgende Verarbeitung wird ~4x schneller ausgeführt und es wird 4x weniger Speicherplatz verbraucht.

Für diejenigen, die technisch versiert sind und eine detailliertere Erklärung wünschen, siehe Punkt „3.7 Eine technische Analyse von Auflösung und Sampling“ unten.

2.2.6 Bessere Daten

Während BlurXTerminator begrenzte Erfassungsfehler – Guidingsfehler, Verkipfung der Fokusebene usw. – teilweise korrigieren kann, gilt immer, dass **bessere** Eingabedaten **bessere** Ausgabebilder erzeugen. Wir verbringen Stunden um Stunden damit, unsere wenigen, kostbaren Photonen zu sammeln. Deshalb lohnt es sich auch in die Verbesserung des bildgebenden Systems z.B. in die Behebung optischer und mechanischer Probleme zu investieren, damit einem qualitativ hochwertigen Ergebnis nicht im Wege steht.

2.2.7 Bilder des Weltraumteleskops

BlurXTerminator wurde nicht auf Sternprofilen von Instrumenten wie den Weltraumteleskopen Hubble oder James Webb trainiert. Es würde daher sehr wahrscheinlich viele Sterne in diesen Bildern nicht als solche erkennen – insbesondere helle Sterne in JWST-Daten – aufgrund ihrer einzigartigen, sechsstrahligen Beugungsmuster. Schwache Sterne werden wahrscheinlich erkannt und können mit den Sternparametern angepasst werden. Nicht-stellares Schärfen kann durchgeführt werden, aber dies sollte mit einem manuellen nicht-stellaren PSF-Durchmesser erfolgen. Die Verarbeitung von Daten dieser Teleskope erfordert eine Anpassung der Parameter.

2.2.8 Bilder mit hohem Dynamikbereich (HDR) unter MacOS

Unter MacOS werden viele neuronale Netzwerkberechnungen unter Verwendung von 16-Bit-Gleitkommaarithmetik durch die von Apple bereitgestellte „CoreML“-Softwarebibliothek durchgeführt. Für die meisten Bilder ist dies eine ausreichende Genauigkeit. In bestimmten Fällen, insbesondere bei Bildern, die Objekte mit extrem hohem Dynamikbereich enthalten – z.B. der Kern von M42, bestimmte Galaxien mit sehr hellen zentralen Kernen usw. – kann dies zu einer leichten Posterisation in den hellsten Bereichen führen. In diesen Fällen können bessere Ergebnisse erzielt werden, indem eine leichte Histogramm-Transformation-Streckung durchgeführt wird, eine *leichte* HDR-Komprimierung (unter Verwendung von HDR-Multiscale-Transform) angewendet wird, um den Dynamikbereich zu reduzieren und dann erst BlurXTerminator angewendet wird. Sterne sollten vor der Anwendung von HDRMT maskiert werden, um eine wesentliche Änderung ihrer Profile zu vermeiden. Das könnte zu Problemen in BlurXTerminator führen.

3. Erörterung

3.1 Wie funktioniert BlurXTerminator?

Alle astronomischen Rohbilder sind das Ergebnis von zwei Dingen: der tatsächlich fotografierten Szene und einer Point-Spread-Funktion (PSF), durch die das Licht dieser Szene eine Convolution erhält oder unscharf wurde.

Jeder Stern in einem Astrofoto, das mit Amateur- (und den meisten professionellen) Geräten aufgenommen wurde, nähert sich einer idealen Punktquelle an: ein Lichtpunkt mit einer Breite von Null, aber mit etwas Helligkeit und Farbe. Wenn wir perfekte Abbildungssysteme hätten – extrem große Optiken, extrem kleine Pixel und keine atmosphärischen Seeing-Effekte, Guidingfehler, optische Verzerrungen usw. – würden alle Sterne in unseren Bildern als winzig kleine Punkte unterschiedlicher Helligkeit und Farbe erscheinen. Stattdessen erhalten wir Sterne mit einem Profil, das eine Breite ungleich Null und eine bestimmte Form hat. Das ideale Punktquellen-Sternprofil wurde durch die Gesamtsumme des Aufnahmeprozesses verändert – verschwommen, also unscharf. Dieser Prozess gilt für *jeden* Lichtpunkt im Bild, nicht nur für die Sterne.

Stellen Sie sich vor, Sie nehmen einen einzelnen Lichtpunkt in der tatsächlichen Szene, sagen wir das Licht eines einzelnen Sterns, und verteilen ihn über mehrere umgebende Kamerapixel, die sich möglicherweise bis in eine große Entfernung von der Position des Punktes erstrecken. Das Licht, das idealerweise nur auf einem der Pixel der Kamera hätte landen sollen, könnte in einem kleinen Teil auf Pixeln landen, die Dutzende, Hunderte oder sogar Tausende von Pixeln entfernt sind. Wiederholen Sie diesen Vorgang für jeden Lichtpunkt in der tatsächlichen Szene auf sehr konsistente und systematische Weise, indem Sie dieselbe Methode zum Ausbreiten des Lichts verwenden. Wenn wir die Ergebnisse dann an jedem Punkt der Szene zu einem zusammengesetzten Bild zusammenfügen, haben wir das, was wir in einem Astrofoto festhalten: eine verschwommene Version der Realität. Technisch gesehen wird die genaue Art und Weise, wie das Licht von jedem Punkt gestreut wurde, als Point-Spread-Funktion (Punktstreuungsfunktion) oder PSF bezeichnet. Das Durchführen dieser Operation an jedem Lichtpunkt in einem Bild wird Convolution (Faltung) genannt. Der Versuch, diesen Vorgang umzukehren, wird Dekonvolution (Entfaltung) genannt.

Die Hauptquellen der Unschärfefunktion, der PSF, sind die endliche Größe unserer Optik und die turbulente Natur unserer Atmosphäre. Die Wellennatur des Lichts, kombiniert mit Optiken begrenzter Größe, führt zu einem Beugungsmuster, der ersten Quelle für Unschärfe. Bezogen auf die

Winkelauflösung am Himmel erzeugen Optiken mit kleineren Durchmessern größere Beugungsmuster, also mehr Unschärfe.

Erdgebundene Teleskope können Licht erst sammeln, nachdem es unsere Atmosphäre passiert hat. Turbulenzen in Kombination mit Temperaturunterschieden in verschiedenen Luftzellen, die das Licht durchdringt, führen zur zweiten Quelle der Unschärfe: dem Seeing.

Das Leben als Astrofotograf wäre einfacher, wenn es hier aufhören würde, aber es gibt viele andere Quellen für Unschärfe. Guidingfehler bringen Streulicht in benachbarte Pixel, oft hauptsächlich in Richtung der Rektaszensionsachse, was zu ovalen Sternen führt. Moderne Teleskopoptiken sind ziemlich gut, aber es können optische Verzerrungen wie chromatische Aberration, Astigmatismus, Bildfeldkrümmung, Koma usw. auftreten. Spikes um helle Sterne herum, die durch Lichtwechselwirkung mit den Fangspiegelstreben des Sekundärspiegels entstehen, erzeugt werden, sind ein Teil der PSF, die sich sehr weit von ihrem Zentrum erstrecken kann. Die "Halos" des Lichts, das Sterne umgibt, wären idealerweise perfekt glatt und symmetrisch, aber kleine Abweichungen in der Optik können in diesen Halos "Strahlen" hervorrufen. Eine ungleichmäßige Behinderung des Lichtwegs durch Strukturen wie Blenden und Objektivblenden kann zu Asymmetrien in den Halos führen.

Jeder dieser Mechanismen trägt zur Gesamt-PSF des Systems bei. Jedes ist eine zusätzliche Möglichkeit, dass das Bild unscharf wird. Es kommt noch schlimmer: Viele dieser Mechanismen sind das, was technisch als *nicht stationär* bezeichnet wird: Sie variieren sogar über das Sichtfeld. Sterne in den Ecken eines Bildes sind beispielsweise selten so scharf wie Sterne in der Mitte.

BlurXTerminator hat zum Ziel, diese Unschärfe mithilfe eines Prozesses namens Dekonvolution umzukehren. Obwohl es einen völlig anderen Ansatz verfolgt, hat es das gleiche Ziel wie klassische Dekonvolutionalgorithmen, wie sie von Richardson, Lucy, van Cittert und anderen entwickelt wurden. Die meisten Implementierungen dieser Algorithmen leiden unter den folgenden Einschränkungen:

- Sie gehen davon aus, dass die PSF über das Sichtfeld hinweg stationär (konstant) ist
- Sie erfordern die Kenntnis der PSF **vor** der Durchführung der Dekonvolution
- Sie erfordern viele Iterationen, um ein Ergebnis zu erzielen
- Sie gehen davon aus, dass die Eingabedaten vollständig linear sind
- Sie gehen von einer PSF in begrenztem Umfang aus

BlurXTerminator hingegen:

- Geht nicht davon aus, dass die PSF stationär ist: Sie kann mit einer PSF umgehen, die über das Feld hinweg variiert
- Erfordert keine *a priori* Kenntnisse der PSF: Es bestimmt die PSF on-the-fly aus den Sternen in einem Bild
- Erzeugt in den meisten Fällen ein Endergebnis in einem Durchgang
- Kann die Nichtlinearität der Detektorsättigung (Clipping) verstehen
- Geht nicht von einer PSF begrenzten Ausmaßes aus

Die letzten beiden Punkte sind sehr wichtig im Umgang mit hellen Sternen. Schwache Sterne wirken sich nur auf eine kleine Anzahl von Pixeln um ihre Schwerpunkte herum signifikant aus. Helle Sterne beginnen, die verlängerten "Flügel" des PSF zu zeigen, was zu größeren (manchmal viel größeren) sichtbaren Halos führt, wenn das Licht von diesen „Flügeln“ heller als der Hintergrund wird. Außerdem sättigen helle Sterne sehr oft den Kamerasensor (Clipping): Die PSF wird oben „abgeschnitten“. Diese Sterne scheinen größere Durchmesser und Halos mit größerer Ausdehnung zu

haben, obwohl ihre Profile in Wirklichkeit von genau der gleichen PSF stammen wie schwächere Sterne.

BlurXTerminator wurde darauf trainiert, mit dieser Situation umzugehen. Es "versteht", dass der tatsächliche Spitzenwert eines hellen, gesättigten Sterns aufgrund von Clipping "außerhalb des oberen Rands" des Bildes liegt. Es wird den Durchmesser dieses Sterns verringern und die Helligkeit und Ausdehnung seines Halo anpassen, als ob das Clipping nicht stattgefunden hätte. Es verarbeitet ein Bild in mehreren Maßstäben gleichzeitig, sodass es Point-Spread-Functions verarbeiten kann, die sich sehr weit von ihren Schwerpunkten erstrecken.

Es ist sinnvoll, sich die Aktion vorzustellen, die BlurXTerminator an einem Bild ausführt, indem es sein vorhandenes PSF in ein anderes (kleineres) PSF übersetzt. Das bekannte Moffat PSF zum Beispiel hat zwei Parameter, Sigma und Beta. Der Sigma-Parameter steuert die Gesamtgröße der PSF, die mit ihrem FWHM-Durchmesser zusammenhängt. Der Beta-Parameter steuert die „Flügel“ der PSF – die schwachen äußeren Ausdehnungen, die hellen Sternen ihre ausgedehnten Halos verleihen. BlurXTerminator bietet separate Steuerelemente, um sowohl den Durchmesser als auch die Halos der stellaren PSF zu modifizieren, wodurch effektive Anpassungen an den Sigma- und Beta-Parametern der PSF unabhängig vorgenommen werden. Das neuronale Netzwerk von BlurXTerminator analysiert die vorhandene PSF im Bild und übersetzt sie in eine andere PSF mit neuen Sigma- und Beta-Parametern, wodurch die PSF effektiv durch eine neue PSF ersetzt wird.

BlurXTerminator führt keine explizite PSF-Extraktion durch – sie wird „on the fly“ während der Verarbeitung durchgeführt. Obwohl es technisch möglich ist, die PSF zu extrahieren, würde dies die neuronale Netzwerkarchitektur verkomplizieren und die Verarbeitung verlangsamen.

3.2 Deconvolution? Wirklich?

Kann ein neuronales Netz tatsächlich eine Dekonvolution im formalen Sinne durchführen?

Kurze Antwort: Ja

Dekonvolution hat eine präzise mathematische Definition. Auf den ersten Blick gibt es keinen einfachen, geschlossenen Ausdruck dafür, welche mathematische Funktion ein neuronales Netzwerk mit mehreren zehn Millionen Parametern und einer Vielzahl nichtlinearer Operationen ausführt. Was jedoch genau ausgedrückt werden kann, ist die Methode, mit der es trainiert wird, und insbesondere, ob diese Methode mit dem universellen Approximationssatz (https://en.wikipedia.org/wiki/Universal_approximation_theorem) übereinstimmt oder nicht, der besagt, dass ein neuronales Netzwerk nachweislich jede kontinuierliche mathematische Funktion mit beliebiger Genauigkeit approximieren kann, sofern bestimmte Voraussetzungen erfüllt sind.

Alle neuronalen Netze werden wie folgt trainiert:

- Geben Sie ihnen einen Input,
- Lassen Sie es mit anfänglich mit zufälligen Gewichtungen arbeiten,
- „bewerte“ ihre Leistung mit Hilfe einer geeigneten *Verlustfunktion*, die das Ausgabeergebnis mit einer gewissen „Grundwahrheit“ vergleicht.
- Verwenden Sie die Kettenregel aus dem Kalkül, den Gradienten der Verlustfunktion in Bezug auf jede Gewichtung im Netzwerk zu berechnen.
- Aktualisieren Sie die Netzwerkgewichtung, um diesen Gradienten zu senken und schließlich
- wiederholen, um die Verlustfunktion unter Verwendung eines geeigneten Optimierungsalgorithmus zu minimieren.

Neuronale Netze werden lernen, das zu tun, wofür sie trainiert wurden. Wenn wir zum Beispiel ein Neuronales Netz trainieren wollten, Astrofotos so aussehen zu lassen, als wären sie z.B. von Van Gogh gemalt worden, würden wir ihm zuerst Amateur-Astrofotos präsentieren, Bilder von Sternenhimmel-Darstellungen als „Grundwahrheit“ verwenden und eine geeignete Verlustfunktion wählen.

Um ein Neuronales Netzwerk darauf zu trainieren, Lösungen für die Dekonvolution anzunähern, müssen wir mit den Trainingsdaten beginnen. Die Trainingseingangsbilder müssen nur durch Dekonvolution mit den „Grundwahrheiten“-Bildern in Beziehung gesetzt werden – dem Ergebnis, das das neuronale Netzwerk zu erzeugen lernen soll. Mit anderen Worten, die "Grundwahrheiten"-Bilder müssen **perfekte** Dekonvolutionen der Eingangsbilder sein. Die Details, wie dies erreicht wird, sind proprietär, aber die Trainingsmethode von BlurXTerminator erfüllt diese Anforderung. Durch Training konvergiert das Netzwerk sogar in Gegenwart von Rauschen ideale Deconvolution- Ergebnisse.

Die verwendete Verlustfunktion ist von entscheidender Bedeutung. Das Training von BlurXTerminator verwendet ausschließlich eine Verlustfunktion, die mit den verfügbaren Beweisen des Theorems übereinstimmt. Dies steht im krassen Gegensatz zu vielen modernen Trainingsmethoden wie den sogenannten „Generative Adversarial Networks“ (GANs). Beim GAN-Training ist die Verlustfunktion die Ausgabe an ein zweites neuronales Netzwerk (des „Diskriminators“), das lernt, die Ausgabe des ersten Netzwerks als „echt“ oder „falsch“ zu deklarieren. Je besser das Hauptnetzwerk darin wird, den Diskriminator zu „täuschen“, desto besser wird der Diskriminator darin, die „Fälschungen“ zu erkennen. „Echt“ und „falsch“ kann alles sein – wofür auch immer der Diskriminator trainiert ist.

Weder diese noch ähnliche generative Methoden wurden verwendet, um die Netzwerkarbeit von BlurXTerminator zu trainieren, da

- 1) ihnen eine rigorose geschlossene Verlustfunktion fehlt, die das Netzwerk dazu bringt, sich einer bekannten mathematischen Funktion anzunähern, und
- 2) sie dem Hauptnetzwerk sehr einfach beibringen können erfinderisch zu werden. Diese letzte Eigenschaft ist in der Tat der Grund, warum GANs in vielen „Style transfer“, der Erzeugung künstlicher Gesichter (<https://thispersondoesnotexist.com>) und ähnlichen Netzwerken verwendet werden, für die *erfinderisches Verhalten* das eigentliche Ziel ist. Sie können fantastisch gut darin sein, Details zu *erzeugen*, und es gibt Hinweise darauf, dass viele beliebte Werkzeuge zum Schärfen von Fotografien mit Methoden wie diesen trainiert wurden, aber das ist nicht das, was wir in einem Dekonvolutionswerkzeug im Astrobereich wollen.

Die letzte Anforderung ist, dass die interne Konstruktion des neuronalen Netzes nur die linearen und nichtlinearen Operationen verwenden darf, die durch die verfügbaren Beweise des Theorems spezifiziert sind. Dazu gehören die "Aktivierungsfunktionen" der einzelnen Knoten innerhalb des Netzwerks. Während die Details wiederum proprietär sind, erfüllen alle Operationen innerhalb des neuronalen Netzwerks von BlurXTerminator diese Anforderungen.

Kurz gesagt, ja, BlurXTerminator erzeugt Ergebnisse, die sich der Dekonvolution mit hoher Präzision annähern. Da alle Implementierungen von Dekonvolutionsalgorithmen grundsätzlich Annäherungen sind, stellt dies BlurXTerminator formal auf eine Stufe mit den klassischen Lösungen für dieses Problem.

3.3 Perfektion?

Klar ist, kein Dekonvolutionstool wird jemals perfekt sein. Wie bereits erwähnt, ist die Deconvolution ein *schlecht gestelltes* Problem und erfordert notwendigerweise Vermutungen seitens des Algorithmus, der die Deconvolution durchführt. Sogar die klassischen Algorithmen werden in Bezug

auf die *Maximum-Likelihood-Schätzungsmethode* diskutiert – sie liefern auch nur *gute Vermutungen* darüber, wie die ursprüngliche Szene aussah. BlurXTerminator versucht dasselbe zu tun.

Im Vergleich zu klassischen Dekonvolutionsmethoden hat ein auf neuronalen Netzwerken basierender Ansatz ein völlig anderes Wesen. Intern verwenden sie einige der gleichen mathematischen Operationen (Convolutions), aber sie sind auch voller nichtlinearer Operationen, die klassische Ansätze eher vermeiden. Dies ist in der Tat ein Teil dessen, was neuronalen Netzwerken ihre Leistungsfähigkeit und Vielseitigkeit verleiht, aber es kann manchmal auch zu seltsamen oder anderweitig falschen Ergebnissen führen.

Ein weiterer wesentlicher Unterschied besteht darin, dass es im Allgemeinen nicht möglich ist, den Grund vollständig nachzuvollziehen, warum ein neuronales Netz unter bestimmten Umständen ein bestimmtes Ergebnis liefert: Neuronale Netze gelten als undurchsichtig. Netzwerke wie das im Herzen von BlurXTerminator führen Hunderte von Millionen bis Milliarden von Berechnungen an einem einzigen Bild durch, wobei sie Dutzende Millionen erlernter Parameterwerte verwenden. Entscheidungen über dieses oder jenes Merkmal oder den Pixelwert werden auf "holografische" Weise über diese Berechnungen verteilt. Den genauen Pfad zu verfolgen, der zu einem bestimmten Ergebnis geführt hat, ist daher ziemlich schwierig und selten nützlich. Der Versuch, Parameterwerte von Hand zu ändern, um in einem Fall ein besseres Ergebnis zu erzielen, führt unweigerlich zu schlechteren Ergebnissen in einer beliebigen Anzahl anderer (unvorhersehbarer) Umstände. Bessere Trainingsdaten und -Methoden sind der Standardweg zur Verbesserung der Leistung eines bestimmten neuronalen Netzwerks.

Große Sorgfalt wurde auf das Design und die Entwicklung der neuronalen Netzwerkarchitektur und insbesondere auf die angewandten Trainingsmethoden verwendet, um sicherzustellen, dass BlurXTerminator Ergebnisse liefert, die der Realität getreu und nicht „erfunden“ sind. Es ist und wird niemals für alle Bilder, die von allen Instrumenten erzeugt werden, in diesem Sinne perfekt sein. Sternschwerpunkte werden möglicherweise nicht perfekt beibehalten. Der Gesamtfluss wird möglicherweise nicht perfekt erhalten. Seine "beste Vermutung" darüber, welche detaillierte Struktur dem unscharfen Originalbild zugrunde liegt, entspricht möglicherweise nicht in allen Fällen der Realität.

Eine Folge davon ist, dass BlurXTerminator niemals für quantitative oder wissenschaftliche Anwendungen verwendet werden sollte. Neuronale Netze und andere Formen des "maschinellen Lernens" finden in der Tat Anwendung in der wissenschaftlichen Astronomie, aber meistens in Form von Werkzeugen, die auf bestimmte Instrumente abgestimmt sind, nicht auf Allzwecklösungen wie BlurXTerminator. Ihre Ergebnisse werden immer noch mit einem etwas voreingenommenen Auge betrachtet, und überraschende oder potenziell wichtige Ergebnisse werden immer mit traditionellen, transparenteren Methoden gegengeprüft.

3.4 Vermeiden Sie eine Überverarbeitung

BlurXTerminator wurde entwickelt, um ein "Überschärfen" bei richtiger Verwendung so weit wie möglich zu vermeiden. Dies ist jedoch dennoch möglich, insbesondere wenn der Parameter PSF-Durchmesser auf einen fiktiven Wert eingestellt oder mehrfach angewendet wird. Es gibt nichts, was dies verhindert, und es kann einige visuell ansprechende Ergebnisse erzeugen, aber es ist keine echte Dekonvolution mehr.

Gleiches gilt für die klassischen Dekonvolutionsalgorithmen, wenn sie beispielsweise mit einer fiktiven Point-Spread-Funktion versorgt werden. Jedes Werkzeug kann für andere als die vorgesehenen Zwecke verwendet werden. Ein Skalpell kann ein präzises, lebensrettendes Werkzeug oder eine Mordwaffe sein, je nachdem, wie es geführt wird.

Die Verarbeitung unserer fotografischen Daten liegt in unseren Händen – es liegt an jedem von uns, ob unsere endgültigen Bilder originalgetreue Darstellungen der Realität oder groteske, überbearbeitete Fantasiebilder sind.

3.5 Zukünftige Verbesserung

Wie andere KI-basierte Tools von RC Astro wird auch das neuronale Netzwerk von BlurXTerminator weiter verbessert. Zu den Zielen für zukünftige Versionen gehören:

- *Sternaureolen*: diffuses Leuchten um sehr helle Sterne, das von einigen Instrumenten oder unter bestimmten Beobachtungsbedingungen erzeugt wird (z. B. dünne Zirkuswolken)
- Feldkrümmung (nicht-stationärer Defokus) – dies kann bereits nach der Anwendung von BlurXTerminator viel weniger auffallen, aber die Feldkrümmung wird derzeit nicht explizit behoben
- Zusätzliche Koma- und Astigmatismusprofile
- MacOS: Bessere/automatische Handhabung von Bereichen mit sehr hohem Dynamikbereich bestimmter Bilder, um gelegentliche leichte Posterisierung aufgrund von Float-16-Arithmetik zu beseitigen
- Umgang mit weißen Bildrändern – dies geschieht manchmal während der Overscan-Kalibrierung und führt derzeit dazu, dass das neuronale Netzwerk überläuft und alle Meßbereiche, die diese weißen Ränder enthalten, einfach komplett weiß darstellt. Um das Problem zu vermeiden können Sie weiße Ränder vorher wegschneiden.

3.6 Support

Unterstützung für BlurXTerminator erhalten Sie auf der Website von RC Astro (<https://www.rc-astro.com/BlurXTerminator/support.php>).

3.7 Eine technische Analyse von Auflösung und Sampling

Über Undersampling und Oversampling in astronomischen Bildern ist viel geschrieben und gesagt worden. Zu diesem Thema bricht regelmäßig eine lebhaftige Debatte aus. In der Tat scheint es derzeit beliebt zu sein, die 2x Drizzle-Integration auf Aufnahmen anzuwenden, die bereits oversampled sind, unabhängig davon, ob dies sinnvoll ist. Es scheint Verwirrung darüber zu herrschen, was überhaupt mit den Begriffen Sampling, Undersampling, Oversampling und vor allem Auflösung gemeint ist. Man vermutet, dass jede Debatte und Meinungsverschiedenheit hauptsächlich auf einen Mangel an Präzision und Klarheit bei der Verwendung dieser Begriffe zurückzuführen ist. Beginnen wir daher damit, sie klar zu definieren:

Auflösung: die minimale Winkelgröße, die von einem Abbildungssystem unterschieden (aufgelöst) werden kann. Die technische Definition dieses Begriffs im Vergleich zu seiner allgemein verwendeten Bedeutung ist eine Hauptursache für Verwirrung – die allgemeine Bedeutung kann uns im Stich lassen, wenn wir versuchen, den Bildgebungsprozess zu verstehen. Zum Beispiel wird oft von Kameras gesprochen, die eine "60-Megapixel-Auflösung" oder ähnliches haben. Die Anzahl der Pixel auf dem Sensor einer Kamera hat *nichts* mit der Winkelauflösung eines bildgebenden Systems zu tun.

Ein gegebenes optisches System hat eine minimale Winkelauflösung, die grundsätzlich durch Beugung begrenzt ist. Atmosphärisches Seeing und die Qualität der Optik können es weiter einschränken. Dies wird vollständig durch die Point-Spread-Function des optischen Systems beschrieben. Wie viel von dieser Auflösung tatsächlich im Bildgebungsprozess erfasst wird, ist eine

Funktion der Pixelgröße des Kamerasensors (nicht seiner Gesamtzahl an Pixeln) im Verhältnis zur Größe der Point-Spread-Function des Systems.

Sampling: Der Prozess der Umwandlung eines Signals, das sich kontinuierlich an jedem Punkt ändert, in einen Satz diskreter Werte oder Samples. In unserem Fall ist das Signal Licht und die Samples sind Pixel. Jede sinnvolle Lichtvariation über die Fläche eines einzelnen Pixels wird in diesem einen Pixel zusammengefasst, und die Information über diese Variation geht damit verloren.

Undersampling: einzelne Samples (Pixel) sind nicht klein genug, um alle sinnvollen Informationen, die in der Point-Spread-Funktion eines Instruments vorhanden sind, angemessen zu sammeln. Beim Undersampling gehen Informationen (feine Details) verloren. Dies ist nicht unbedingt eine Katastrophe: Oft wird ein Kompromiss eingegangen, vielleicht zwischen der Auflösung einerseits und dem Sichtfeld oder der verfügbaren/erschwinglichen Kameratechnologie andererseits.

Oversampling: Einzelne Pixel sind so klein dass sie alle sinnvollen Variationen in der Point-Spread-Funktion eines Instruments mehr als notwendig erfassen. Beim Oversampling gehen natürlich keine Informationen verloren, aber übermäßiges Oversampling fügt auch keine sinnvollen Informationen hinzu: Es verbraucht nur Speicherplatz, erhöht die Bildverarbeitungszeit und belastet die Algorithmentwickler unnötig.

Wie unten gezeigt wird, gibt es keine scharfe Trennlinie zwischen Undersampling und Oversampling. Wenn ein Bild leicht undersampled ist, gehen einige feine Informationen verloren, aber die Bildqualität „fällt nicht über die Klippe“ wie man so schön sagt. Wenn die Wahlmöglichkeit besteht, ist es besser, leicht oversampled als leicht undersampled zu sein.

Der Rest dieses Abschnitts ist eine einfache Analyse unter Verwendung grundlegender Signalverarbeitungskonzepte (<https://de.wikipedia.org/wiki/Signalverarbeitung>), die dieses Thema hoffentlich weiter beleuchten werden. Dies sollte für jeden mit einer angemessenen Grundlage in der Signalverarbeitungstheorie einfach zu verstehen sein.

Für diejenigen, die diesen Hintergrund nicht haben, hier sind die Schlussfolgerungen:

- Bilder mit PSF-FWHM-Werten (Sterndurchmesser) von 4 Pixeln sind großzügig gesampled).
- Bilder mit 4-8 Pixel FWHM sind oversampled, aber wahrscheinlich nicht genug, um ein Downsampling (Heruntersamplen) um den Faktor 2 zu rechtfertigen, es sei denn, sie sind ziemlich verrauscht.
- Bilder mit mehr als 8 Pixel FWHM sind mehr als 2x oversampled und können ohne signifikanten Informationsverlust um den Faktor 2 heruntergesampelt werden. IntegerResample im Average-Modus sollte verwendet werden, um das Aliasing von Rauschen zu minimieren und das SNR im heruntergerechneten Bild zu maximieren.
- Die Durchführung einer 2x Drizzle-Integration von Subframes zur Wiederherstellung der Auflösung ist nur bei undersampleten Daten sinnvoll (FWHM-Werte kleiner als 4 Pixel). 1x Drizzle-Integration von oversampleten Daten kann sinnvoll sein, um die Interpolationsartefakte zu eliminieren, die von anderen Registrierungsmethoden erzeugt werden, vorausgesetzt, es gibt genügend Subframes mit ausreichendem Dithering, um Drizzle-Artefakte zu vermeiden.

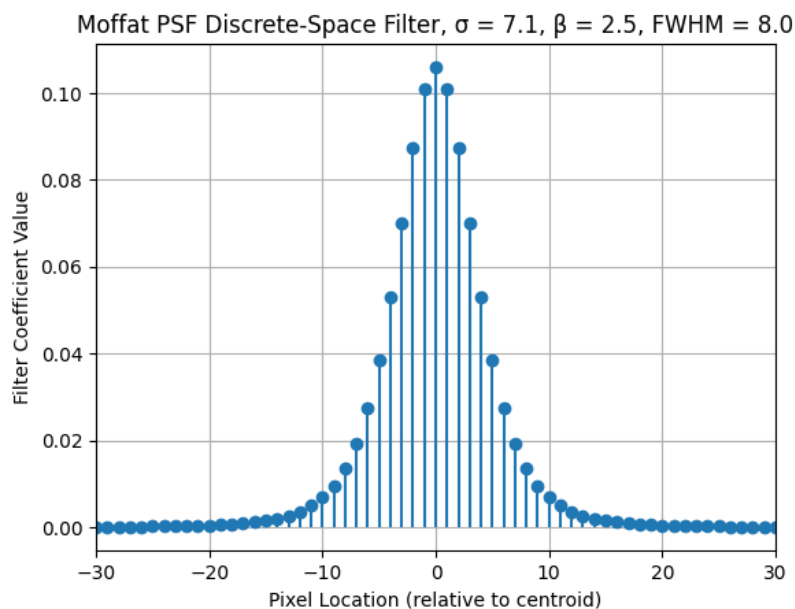
Beachten Sie, dass diese Schlussfolgerungen nicht von der Öffnung der Optik, der Brennweite, dem Öffnungsverhältnis, der Pixelgröße oder dem Seeing am Beobachtungsort abhängen. Diese sind wichtig bei der Auswahl der Komponenten eines Abbildungssystems und der Wahl des Standorts,

aber sobald ein System aufgebaut ist, wird es jede PSF erzeugen, die es erzeugen wird, und diese PSF wird von den Pixeln der Kamera abgetastet. Die nachstehende Analyse zeigt, dass die erreichbare Winkelauflösung eines gegebenen Abbildungssystems durch die Point-Spread-Funktion und **nicht** durch die Pixelgröße begrenzt wird, sobald 4 oder mehr Pixel die FWHM der PSF überspannen.

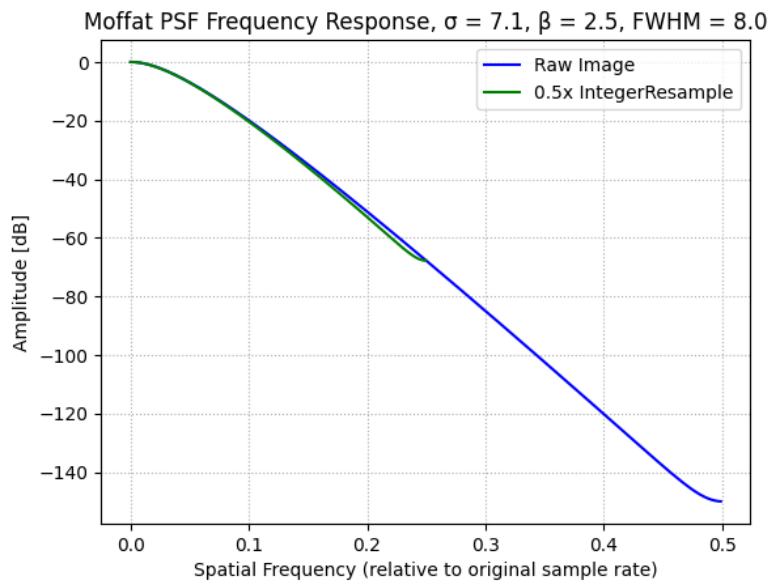
3.7.1 Analyse von Auflösung und Sampling unter Verwendung von Signalverarbeitungskonzepten

Unsere Bilder sind das Ergebnis der Abtastung eines Signals, das mit einer Point-Spread-Funktion gefaltet wurde. Dies lässt sich leicht als zeitdiskretes (oder eher raumdiskretes) Filter modellieren, von dem aus wir den Prozess in Bezug auf die vertrauten Signalverarbeitungskonzepte von Frequenzgang und Aliasing analysieren können.

PSFs, die typischerweise in astronomischen Bildern anzutreffen sind, werden sehr gut durch eine Moffat-Verteilung dargestellt. Hier ist ein Diagramm einer eindimensionalen, abgetasteten Moffat-PSF, die als Diskreter-Raum-Filter betrachtet wird. Dieser Filter hat Moffat-Verteilungsparameter (https://en.wikipedia.org/wiki/Moffat_distribution), die so gewählt sind, dass er einen FWHM-Durchmesser von 8 Pixeln hat.



Hier ist der räumliche Frequenzgang dieses Filters zusammen mit dem effektiven 0,5-fachen Integer-Resample-Frequenzgang, der nur eine Zwei-Sample-BoxCar-Mittelung ist, gefolgt von einer Dezimierung. Diese Antwort wird auf die ursprüngliche Abtastrate bezogen, um den Vergleich zu erleichtern:



Wie zu sehen ist, beträgt die Dämpfung bei der Nyquist-Ortsfrequenz (<https://de.wikipedia.org/wiki/Nyquist-Shannon-Abtasttheorem>) durch diese PSF im Rohbild über 140 dB, wodurch jedes Signal bei dieser Frequenz *weit* unter jedem vorstellbaren echten Grundrauschen liegt. Selbst bei der halben Nyquist-Frequenz beträgt die Dämpfung etwa 70 dB: Informationen bei dieser Ortsfrequenz wurden um einen Faktor von etwa 3.000 gedämpft. Dies kann als vernünftiges Anti-Aliasing-Filter zur Dezimierung um den Faktor zwei angesehen werden. (Anti-Aliasing von Rauschen zwischen den halben Nyquist- und Nyquist-Frequenzen wird effektiv durch die Mittelungsoperation in Integer-Resample erreicht.)

Das Downsampling eines Bildes mit dieser PSF um 2x hat keine signifikante Auswirkung auf das tatsächlich wiederherstellbare Signal, das in dem Bild vorhanden ist und es wird kein signifikantes Aliasing auftreten. Ja, es ist theoretisch möglich, ein Bild mit einem so hohen SNR zu haben, dass ein geringfügiger Informationsverlust oder Aliasing auftreten würde, aber praktisch betrifft das uns eher nicht: Wir können leider nur davon träumen, dass unsere Rohbilder so rauschfrei wären.

Die obigen Dämpfungswerte variieren ein wenig mit dem Beta-Parameter des Moffat PSF, aber nicht genug, um bei dieser Schlussfolgerung einen Unterschied zu machen. Selbst bei einem extremen Beispiel einer Lorentzischen PSF (entspricht Moffat mit Beta von 1,0) beträgt die Nyquist-Dämpfung 100 dB und die halbe Nyquist-Dämpfung etwa 55 dB.

Aus diesem Grund beträgt der maximale PSF-Durchmesserwert in BlurXTerminator 8 Pixel. Jedes Bild mit Sternen mit FWHM-Werten von mehr als 4 Pixeln kann bereits als oversampled betrachtet werden. Das Zulassen von bis zu 8-Pixel-FWHM-Werten bietet 2-fachen Overhead für ganzzahliges Resampling ohne signifikanten Informationsverlust.

3.8 GPU-Beschleunigung

Unter MacOS erfolgt die Verwendung einer GPU (oder „neuronalen Engine“ auf Macs auf Apple-Siliziumbasis) zur Beschleunigung neuronaler Netzberechnungen automatisch und wird von der von Apple bereitgestellten „CoreML“-Bibliothek gehandhabt. Diese Bibliothek trifft die Entscheidung, ob eine bestimmte Hardwarekonfiguration zur Beschleunigung des neuronalen Netzwerks von BlurXTerminator verwendet werden kann oder nicht. Die meisten Macs der letzten Jahre profitieren von der Beschleunigung, ohne dass eine zusätzliche Konfiguration erforderlich ist.

Auf Windows- und Linux-Computern werden die neuronalen Netzwerkberechnungen mithilfe der von Google bereitgestellten TensorFlow-Bibliothek durchgeführt. Eine einzelne CPU-Version davon wird standardmäßig mit PixInsight auf Hardware installiert, die dies unterstützt. Wenn Ihr Computer über eine kompatible NVIDIA-GPU verfügt, können BlurXTerminator und andere auf neuronalen Netzwerken basierende Tools möglicherweise erheblich beschleunigt werden.

Leider macht es NVIDIA kleinen Entwicklern nicht leicht, alle zusätzlichen Softwarebibliotheken zu lizenzieren, die dazu erforderlich sind. Das Bewerkstelligen der Beschleunigung ist daher leider eine komplexe Aufgabe, die verschiedene Downloads und Installationen, das Setzen von Umgebungsvariablen usw. beinhaltet. Noch verwirrender kann es werden, wenn man bedenkt, dass die Versionen aller heruntergeladenen Komponenten miteinander kompatibel sein müssen.

Für diejenigen, die über die technischen Fähigkeiten verfügen und sich dieser Aufgabe gewachsen fühlen, finden hier eine kurze Anleitung für Windows-Computer. Dies ist eine vereinfachte Version von William Lis ausgezeichnetem Leitfaden zur Beschleunigung von StarNet (<https://www.williamliphotos.com/starnet-cuda>) . Ähnliche Anweisungen werden in diesem NVIDIA-Handbuch (<https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html>) behandelt, das auch Anweisungen für Linux-Computer enthält.

Wenn Sie Zweifel haben, ob Sie diese Schritte erfolgreich durchführen können, suchen Sie sich einen technisch versierten Freund, der Ihnen hilft. Durch dieses Verfahren wird wahrscheinlich der Grafiktreiber für Ihre GPU aktualisiert. Wenn dies andere Anwendungen beeinträchtigen könnte, die von einer bestimmten Grafiktreiberversion abhängen, sollten Sie Ihr System sichern, damit Sie es bei Bedarf wiederherstellen können.

3.8.1 Kompatibilität

Sie benötigen ein Intel/AMD x64-System mit Windows 10 oder höher und eine NVIDIA-GPU mit CUDA-Rechenfunktionen der Version 3.5 oder höher. GPUs mit weniger als 2 GB integriertem RAM sind möglicherweise nicht ausreichend. Überprüfen Sie die NVIDIA-Seite, um die Fähigkeiten Ihrer GPU zu überprüfen.

Sie benötigen außerdem Administratorrechte für Ihr Benutzerkonto, um viele der Änderungen vorzunehmen. Windows wird wahrscheinlich um Erlaubnis bitten, eine Reihe von Aktionen auszuführen.

3.8.2 Laden Sie das NVIDIA CUDA-Toolkit herunter und installieren Sie es

„CUDA“ steht für **Compute Unified Device Architecture**, NVIDIAs Name für eine Reihe von Softwarebibliotheken, die es ermöglichen, auf vielen ihrer Grafikprozessoren (GPUs) allgemeines Computing durchzuführen. Das CUDA-Toolkit-Installationsprogramm kann von der NVIDIA-Seite heruntergeladen werden (<https://developer.nvidia.com/cuda-downloads>) . Wählen Sie Windows, x86_64, Ihre Windows-Version, „exe (lokal)“ und klicken Sie schließlich auf die Download-Schaltfläche. Führen Sie das Installationsprogramm aus und wählen Sie Express-Installation, um alle Komponenten zu installieren.

Dadurch wird auch der Grafiktreiber für Ihre GPU aktualisiert und sichergestellt, dass er mit der Version des CUDA-Toolkits im Download kompatibel ist.

Der Installer sollte auch eine Reihe von Umgebungsvariablen setzen, die später benötigt werden. Die Dateien, aus denen das Toolkit besteht, sollten an einem Speicherort wie C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\bin. installiert werden. Das letzte Bit ist die Versionsnummer des Toolkits. Die neueste davon ist Version 11.8 zum Zeitpunkt des Schreibens dieses Artikels.

3.8.3 cuDNN-Dateien herunterladen und installieren

Die Abkürzung „cuDNN“ bezieht sich auf noch mehr Softwarebibliotheken, die für die Beschleunigung von „Deep Neural Network“-Berechnungen auf CUDA-fähigen Geräten sorgen. Das Herunterladen dieser Komponente erfordert ein NVIDIA-Entwicklerkonto, das kostenlos erstellt werden kann.

Die cuDNN-Bibliotheken können von der NVIDIA-Seite (<https://developer.nvidia.com/rdp/cudnn-download>) heruntergeladen werden. Nachdem Sie ein Konto erstellt und Ihre E-Mail-Adresse verifiziert haben, sollte Ihnen eine Liste mit Installationsprogrammen für verschiedene Betriebssysteme angezeigt werden. Klicken Sie auf den Link „Local Installer for Windows (Zip)“.

Obwohl dies als Installationsprogramm gekennzeichnet ist, ist dies nicht der Fall. Es ist eine Sammlung von Dateien in einem komprimierten Archiv, von denen nur einige benötigt werden und die manuell an den gewünschten Ort kopiert werden müssen. Suchen Sie im ZIP-Archiv den Ordner bin. Kopieren Sie den Inhalt dieses Ordners in den bin-Ordner in der CUDA-Toolkit-Installation von oben, z. B. C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\bin.

Dateien aus den Ordnern lib oder include werden nicht benötigt, es sei denn, Sie entwickeln tatsächlich Ihre eigene GPU-beschleunigte neuronale Netzwerksoftware.

3.8.4 Laden Sie die ZLIB-Komprimierungsbibliothek herunter und installieren Sie sie

Es gibt eine Softwarebibliothek zur Datenkomprimierung, von der die oben genannten Bibliotheken für einige Operationen abhängig sind. Es kann hier heruntergeladen werden. Dekomprimieren Sie das heruntergeladene Archiv und suchen Sie den Ordner dll_x64 darin. Kopieren Sie die Datei mit dem Namen **zlibwapi.dll** wie oben in das bin-Verzeichnis des CUDA-Toolkits.

3.8.5 Laden Sie die GPU-fähige TensorFlow-Bibliothek herunter und installieren Sie sie

Das TensorFlow-Projekt verwaltet verschiedene Versionen einer Softwarebibliothek namens **tensorflow.dll**. Es ist diese Bibliothek, die BlurXTerminator und andere auf neuronalen Netzwerken basierende Tools verwenden, um Berechnungen durchzuführen. Die GPU-fähige Version der tensorflow.dll-Bibliothek wiederum hängt von den oben installierten CUDA- und cuDNN-Bibliotheken ab.

Die mit PixInsight installierte Version von tensorflow.dll unterstützt nur CPU-Operationen. Eine Version, die GPU-Beschleunigung unterstützt, kann von der TensorFlow-Seite heruntergeladen werden (https://www.tensorflow.org/install/lang_c).

- Suchen Sie nach dem Eintrag „Windows GPU only“ und laden Sie das ZIP-Archiv herunter.
- Dekomprimieren Sie es und suchen Sie im lib-Ordner nach der Datei „tensorflow.dll“.
- Suchen Sie den bin-Ordner von PixInsight auf Ihrer Festplatte, normalerweise C:\Programme\PixInsight\bin
- Benennen Sie die dort gefundene Datei „tensorflow.dll“ in etwas wie tensorflow_cpu.dll um. Dies ist die reine CPU-Version der TensorFlow-Bibliothek, die mit PixInsight vertrieben wird. Wenn Sie es umbenennen, anstatt es zu ersetzen, können Sie leicht darauf zurückgreifen, wenn etwas schief geht.

- Verschieben Sie die neue „tensorflow.dll“-Datei, die heruntergeladen wurde, in den bin-Ordner von PixInsight.

3.8.6 Verifizieren/Setzen von Umgebungsvariablen

Die Installation des obigen CUDA-Toolkits sollte einige Umgebungsvariablen gesetzt haben, die benötigt werden, damit die tensorflow.dll-Bibliothek alle Zugaben der GPU-Beschleunigung finden kann. Überprüfen Sie diese Umgebungsvariablen wie folgt:

- Starten Sie den Editor für Windows-Umgebungsvariablen in der Systemsteuerung
- Unter „Systemvariablen“ sollte eine CUDA_PATH-Variable auf C:\Programme\NVIDIA GPU Computing Toolkit\CUDA\v11.8 gesetzt sein, mit möglicherweise einer anderen Versionsnummer, je nachdem, was Sie oben installiert haben.
- Es sollte auch eine CUDA_PATH_V11_8-Variable (oder eine ähnliche, abhängig von Ihrer CUDA-Toolkit-Version) geben, die auf denselben Speicherort verweist.
- Die Systempfadvariable sollte die Ordner bin und libnvvp des CUDA-Toolkits enthalten: C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\bin und C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8\ libnvvp.

Eine zusätzliche Umgebungsvariable sollte festgelegt werden, um die TensorFlow-Bibliothek anzuweisen, nur so viel GPU-Speicher zuzuweisen, wie erforderlich ist, im Gegensatz zu ihrem Standardverhalten, das darin besteht, den gesamten GPU-Speicher zuzuweisen:

- Erstellen Sie eine neue Systemumgebungsvariable mit dem Namen TF_FORCE_GPU_ALLOW_GROWTH und setzen Sie sie auf TRUE

Überprüfen Sie nach dem Schließen des Umgebungsvariablen-Editors, ob die Änderungen wirksam wurden. Starten Sie eine DOS-Eingabeaufforderung und führen Sie den Befehl **set** aus. Dies listet alle aktuell eingestellten Umgebungsvariablen und ihre Werte auf. Wenn Sie die vorgenommenen Änderungen nicht sehen, müssen Sie Ihren Computer möglicherweise neu starten, damit sie wirksam werden.

3.8.7 Genießen Sie die schnelle neuronale Netzwerkverarbeitung

Das sollte die Einrichtung zur Beschleunigung neuronaler, netzbasierter Berechnungen vervollständigen – starten Sie PixInsight und genießen Sie es. Wenn Sie BlurXTerminator zum ersten Mal auf einem Bild ausführen, kann es anfangs länger dauern, bis es losgeht – diese Verzögerung ist auf die oben genannten Softwarebibliotheken zurückzuführen, die die Operationen des neuronalen Netzwerks auf hochgradig parallele GPU-Hardwareoperationen abbilden. Nach dieser anfänglichen Verzögerung sollte die Verarbeitung viel, viel schneller als zuvor ablaufen.

Alle anderen Anwendungen oder Plug-Ins auf Ihrem Computer, die TensorFlow zum Ausführen neuronaler Netze verwenden, sollten ebenfalls beschleunigt werden können, indem die Instanzen der Datei tensorflow.dll ersetzt werden, die sie laden.