Kometenbearbeitung mit PixInsight Tobias Lindemann 12.06.2020 V3.3





Diese Anleitung ist angelehnt an das Videotutorial von Edoardo Luca Radice, das unter folgendem Link zu finden ist:

https://www.youtube.com/watch?v=RmgjvC8c_yc

Die Beispielbilder während der Bearbeitung enthalten verschiedene Kometen, die ich im Laufe der Jahre aufgenommen habe.

Überblick

- 1. Bilder wie gewohnt auf Sterne registrieren
- 2. Vor der Weiterverarbeitung Normalisieren (LocalNormalization)
- 3. Bilder auf Kometen ausrichten und integrieren ->Kometenbild wird erstellt
- 4. Kometenbild anpassen
- 5. Ergebnis von Ausgangsbildern abziehen, Bild ohne Komet wird erstellt
- 6. Ergebnis von Ausgangsbildern abziehen
- 7. Bilder integrieren, verbessertes Kometenbild wird erstellt
- 8. Bilder zusammenfügen

Bearbeitung:

1. Bilder auf Sterne registrieren

Bilder kalibrieren, evtl. debayern und ein Staralignment ausführen. Dies kann sowohl manuell, als auch mit BPP/WBPP passieren. Die Integration muss aber erst später ausgeführt werden. Bei der Kometenbearbeitung ist es besonders wichtig, dass die Ausgangsbilder möglichst wenig Hotpixel enthalten. Falls nötig sollte man vorher eine CosmeticCorrection durchführen. Mit dem Blinkmodul kann man die Bilder überprüfen. Die Sterne sollten nun still stehen und der Komet durch das Feld ziehen.



http://ffmpeg.org/download.html Da die Qualität mit ffmpeg aber zu Wünschen übrig lässt, kann man auch Einzelbilder erzeugen und später mit einer anderen Videosoftware bearbeiten.

2. Bilder normalisieren

Da Kometen in ihrer besten Phase oft nur kurz nach Sonnenuntergang oder vor Sonnenaufgang sichtbar sind, ändert sich die Himmels-Helligkeit in Laufe der Sequenz oft stark. Damit wir später den Kometen von jedem Einzelbild korrekt abziehen können, sollte aber die Helligkeit so gut es geht gleich bleiben. Dafür benutzen wir LocalNormalization. Um ein gutes Referenzbild auszusuchen nehmen wir Blink. Es sollte ein möglichst dunkles Bild sein und wenn möglich keine großen Satellitenspuren o.ä. enthalten. Keinesfalls sollten Wolkenzierren im Bild zu sehen sein. Wir wählen also das bestmögliche Bild als Referenz und laden alle Bilder mit "Add files". Normalerweise gibt das Tool keine normalisierten Bilder aus, sondern schreibt nur Dateien, die die mathematischen Funktionen für die Normalisierung enthalten – sogenannte XMLN-Dateien. Diese werden dann später mit dem Tool ImageIntegration verrechnet. Weil wir aber tatsächlich normalisierte Bilder benötigen, müssen wir bei "Apply normalization" Always wählen. Damit werden sowohl normalisierte Bilder, als auch die XMLN-Dateien ausgegeben. Wir benötigen diese Dateien nicht, deshalb können wir den Haken bei "generate normalization data" deaktivieren.



Da sich der Komet in der Sequenz bewegt, würde er auch bei der LocalNormalization verrechnet werden und die Bilder würden an den Kometen des Referenzbildes angeglichen werden. Um das zu vermeiden, sollten wir bei "Scale" einen möglichst großen Wert angeben. Damit werden nur sehr großflächige Unterschiede angeglichen. Der Wert sollte ein Vielfaches von 32 und maximal halb so groß wie die Bilddimensionen sein.

3. Komentenregistrierung

Den Prozess CometAlignment aufrufen. Die kalibrierten, registrierten und normalisierten Bilder laden. Im Feld "Parameters", den ersten Show-Button drücken. Nun öffnet sich das erste Bild in der Reihe. Nachdem AutoSTF (STRG-a) angewandt ist, hier den Kometenkern im Bild wählen. Dann den zweiten Show-Button drücken, der das letzte Bild in der Reihe öffnet. Auch hier den Kometenkern wählen. Die Kometenposition der mittleren Bilder werden durch den Zeitstempel automatisch berechnet, der in den Bildern bei der Aufnahme gespeichert wurde. Das Referenzbild sollte eines der mittleren Bilder in der Serie sein. Mit einem Doppelklick auf die Bildnummer (#), wählen wir es aus. Es wird mit einem blauen Pfeil angezeigt. Da wir den Prozess öfter verwenden und das Referenzbild immer das selbe sein muss, ist es auch gut sich den Dateinamen zu merken, oder aufzuschreiben. Jetzt wählen wir noch ein Output directory. Den Rest belassen wir zuerst unangetastet. Nachfolgend lassen wir den Prozess mit global apply (ausgefüllter blauer Kreis) ablaufen. Wichtig: Wenn der Prozess durch ist, CometAlignment nicht schließen, sondern mit dem Dreieckchen auf die Arbeitsfläche ziehen. Den Prozess brauchen wir später nochmal.



Wenn das Icon auf der Arbeitsfläche abgelegt ist, können wir den aktuellen Prozess (CometAlignment) und die beiden Bilder schließen.

Mit dem Blinkmodul können wir anschließend prüfen, ob die eben erstellten Bilder richtig registriert wurden. Nun sollte der Komet still stehen und die Sterne durch das Bild wandern.



In der Sequenz sind die Sterne also in jedem Bild an einer anderen Stelle. Das können wir beim nächsten Schritt ausnutzen. Als nächstes öffnen wir den Prozess ImageIntegration und laden die eben erstellten Bilder mit "Add Files". Wir wollen nun ein Bild erzeugen, das nur den Kometen enthält. Da die Sterne ja in jedem Bild an einer andern Stelle sind, werden sie durch die Pixel Rejection weitgehend entfernt. Wir wählen zunächst mal Winsorized Sigma bei Standard Sigma-Werten. Diese müssen wir aber später noch anpassen. Da wir im nachfolgenden Schritt das entstandene Kometenbild von den Ausgangsbildern abziehen wollen, dürfen wir keine Normalization wählen. Durch die Normalization würde das Bild auf ein anderes Level gehoben werden und sich von den Ausgangsbildern unterscheiden. Auch bei Weights tragen wir "Don't care" ein. Es kommt dabei noch nicht darauf an, das beste Signal-Rauschverhältnis zu erhalten, sondern es sollten alle Bilder gleichmäßig in das Ergebnis mit einfließen. Nachdem die Integration abgeschlossen ist, den Prozess nicht schließen, denn das Ergebnis wird noch nicht optimal sein und wir müssen die Rejection noch anpassen. Dadurch, dass die Bilder auf den Kometen registriert wurden, entstanden schwarze Ränder. Damit diese keinen Einfluß auf das Ergebnis haben, müssen wir den Haken bei ",Clip low range" setzen.



Das Ergebnis sollte dann etwa so aussehen:



Der Komet ist schon ganz gut, nur die Reste der Sterne stören. Mit einer härteren Pixel Rejection kann man diese noch weiter reduzieren. Auch ein anderer Rejection Algorithmus kann ein besseres Ergebnis zeigen. Wenn man ausreichend viele Bilder (>20) zur Verfügung hat, erhält man mit Linear Fit oft das beste Ergebnis. Wenn große/helle Sterne im Bild sind, kann auch eine "Large-scale Pixel Rejection" helfen. Den richtigen Algorithmus und Wert für Sigma high zu finden ist nicht ganz einfach und erfordert mehrere Versuche. Damit das Ganze schneller geht, können wir uns ein Preview im Bild ziehen und nur diesen Teil integrieren. Das spart viel Zeit. Am besten wählt man einen Teil, der den Kometen enthält und ganz hartnäckige Sternspuren. Ganz unten im Prozessfenster gibt es den Abschnitt "Region of Interest". Dort können wir mit dem Button Preview unseren Ausschnitt wählen. Damit mehr Sternspuren entfernt werden, müssen wir "Sigma high" reduzieren. Wir gehen mal auf 1.5 und schauen uns das Ergebnis an.

3 1:5 integration <*new*>	- - + ×	Σ ImageIntegration	ж×
		Input Images	\$
		1 🗸 _M3G3509_c_d_r_ca.xisf	Add Files
		2 ✓ _M3G3510_c_d_r_ca.xisf	Add L.Norm, Files
		3 ✓ _M3G3511_c_d_r_ca.xisf	Clear L Norm Files
		4 ✓ _M3G3512_c_d_r_ca.xisf	Add Drivela Silan
		5 / _M3G3513_C_0_r_C8.XISF	Add Drizzle Files
Preview01		7 ✓ _M3G3515_c_d_r_ca.xisf	Clear Drizzle Files
		8 🗸 _M3G3516_c_d_r_ca.xisf	Set Reference
		9 ✓ _M3G3517_c_d_r_ca.xisf	Select All
		10 V _M3G3518_c_d_r_ca.xisf	Invert Selection
		12 / M3G3520 c d r ca xisf	Toggle Selected
		13 ✓ _M3G3521_c_d_r_ca.xisf	Remove Selected
		14 🗸 _M3G3522_c_d_r_ca.xisf	Clear
		15 🗸 _M3G3523_c_d_r_ca.xisf	Static data targets
		16 ✓ _M3G3524_c_d_r_ca.xisf	Full paths
		Format Hints	¥
		Image Integration	¥
		Pixel Rejection (1)	¥
		Pixel Rejection (2)	\$
		Min/Max low: 1	
		Min/Max high:	
		Percentile low: 0.200	
		Percentile high: 0 100	
		Sigma low: 4 000	0
		Ciama histo 1 500	~
		Winnerientien erteff. 5 000	
		winsonzation cutorr: 5.000	
		Linear fit low: 5.000	
		Linear fit high: 4.000	
		ESD outliers: 0.30	
		ESD significance: 0.05	
		ESD low relaxation: 1.50	
		Range low: 0.000000	
rejection high		Range high: 0.980000	
rejection_night w		Pixel Rejection (3)	¥
		Large-Scale Pixel Rejection	ScreenTransferi a ▼
Tejection_low		Region of Interest	\$
		Left: 1945	
		Top: 750	
		Width: 2475	
- x x		Height: 1330	From Preview
		1330 V	4.155 MLS

Nach ein paar Tests mit verschiedenen Sigma high-Werten, kann man sich den Unterschied anschauen und muss abwägen zwischen guter Rejection und noch gutem Signal-Rauschverhältnis. Je kleiner der Sigma-Wert ist, umso besser ist die Rejection, aber auch das Rauschen nimmt zu. In meinem Fall entscheide ich mich für den Wert 1.2, da der Unterschied zu niedrigeren Werten nur noch marginal ist und noch ein gutes SNR herrscht. Ein guter Anhaltspunkt ist auch die Rejection_high Map. Wenn diese fast nur noch helle Pixel enthält, ist die Ausreißer-Statistik zu hart und das Rauschen nimmt stark zu. Den Sigma low kann man auf dem Standard-Wert belassen, denn es sollen nur helle Ausreißer(Sterne) aus dem Bild fliegen.



Der Wert ist nun ermittelt, daher können wir den Haken bei "Region of Interest" deaktivieren und führen den Prozess mit dem richten Sigma high nochmals aus.

4. Kometenbild anpassen

Da nun immer noch einige Reste von Sternspuren vorhanden sind, können wir zumindest diese glätten, die keinen Kometen enthalten. *Hinweis: Es wird dabei nicht das Endergebnis geglättet, sondern nur ein Bild, das wir später von den Einzelbildern abziehen.*

Für den glatten Hintergrund benutzen wir das Tool DBE. Allerdings entfernen wir damit nicht die Gradienten, sondern erzeugen uns damit ein Modell, das unserem Hintergrund gleicht. Dazu wählen wir wie gewohnt manuell Samples, in denen weder Sternspuren, noch sonstige Fehler zu sehen sind. Die Samples sollten nur Hintergrund enthalten. Bei Correction wählen wir "none", weil wir keine Korrektur durchführen wollen, sondern nur ein Hintergrundmodell erzeugen möchten. Damit das Modell gleich groß, wie unser Bild ist, müssen wir bei Downsampling "1" eingeben.



Das entstandene Hintergrundmodell benennen wir um in G und speichern es ab. Wir können es zum Test auch mal über unser Kometenbild setzten und mit STRG-Bild runter die Bilder blinken. Wenn das gleiche STF angewandt ist, sollte sich die Helligkeit des Hintergrunds nicht ändern.



Dieses Hintergrundmodell wollen wir nun überall dort einsetzten, wo der Komet nicht ist. Dazu müssen wir für das Kometenbild eine Maske erzeugen. Seit kurzem (Juni 2020) gibt es eine neue Version eines Skript (GAME v1.51), mit dem man recht einfach manuell Masken erstellen kann. Wie man zu diesem Skript (und vielen weiteren) kommt, ist auf der Seite Skypixels.at beschrieben.



Man startet also das Skript GAME, wählt "Multi point" und startet mit "add" eine neue Auswahl. Diese sollte großzügig den Kometenkern und den Schweif enthalten. Um eine weiche Maske zu erhalten wählen wir noch "Gradient Edge Mask" und starten mit OK.

Mit den Prozess Convolution kann man die Maske noch etwas weicher zeichnen.

Nun erstellen wir eine Kopie des Kometenbildes und ziehen die neu erstellte Maske auf diese Kopie. Dabei kann man nochmal prüfen, ob die Maske den Kometen komplett abdeckt.



Damit der Komet rot abgedeckt wird, müssen wir die Maske invertieren. Jetzt öffnen wir Pixelmath und geben in die erste Zeile folgenden Term ein: med(\$T)+G-med(G). Erklärung: \$T= aktuelles Bild, med=Median(Mittelwert). Wir wählen also den Mittelwert des aktuellen Bildes und addieren das Bild G hinzu, ziehen aber von diesem wieder den Mittelwert ab. Das hat die Auswirkung, dass sich an der mittleren Helligkeit des Zielbildes nichts ändert. Dann ziehen wir das kleine Dreieckchen auf unser maskiertes Bild. Nun sollte unser Hintergrundmodell G überall auf dem Kometenbild liegen, nur nicht auf dem Kometen selbst. Die Helligkeit sollte sich nicht geändert haben. Das ist wichtig, damit wir den Kometen korrekt von den Rohbildern abziehen können. Jetzt speichern wir das Bild als Komet1mod. In diesem Bild ist also der Teil, der nicht den Kometen enthält, mit dem Hintergrundmodell gefüllt. Der Teil, der den Kometen enthält, ist so gut es geht von Sternen befreit.

5. Ergebnis von Ausgangsbildern abziehen, Bild ohne Komet wird erstellt

Als nächstes öffnen wir wieder den gespeicherten Prozess CometAlignment, den wir als Icon abgelegt haben. Dort sollten noch alle Rohbilder mit den gespeicherten Werten enthalten sein. Operand image ist dieses mal unser modifiziertes Kometenbild "comet1mod", da wir dieses abziehen wollen. Bei "Operand is" wählen wir "Comet aligned" und bei Enable LinearFit deaktivieren wir das Feld. Die Funktion Linear Fit soll hier das Kometenbild an die Einzelbilder angleichen. Das hilft, wenn zuvor bei der Integration eine Normalisierung angewandt wurde. Da wir das aber nicht gemacht haben, brauchen wir auch kein LinearFit. Ohne Normalisierung und LinearFit wird das Ergebnis genauer.

Subtract		\$
Operand image:	G:/2019Astro/20190209Iwamoto1D/Bearbeitung2020/masters/comet1mod.xisf	≣ €3
	— Operand Origin: — — Operand is: — — — Drizzle save: —	
	ImageIintegration Stars aligned Stars aligned	
	O DrizzleIntegration	
	Enable LinearFit 🗹 Normalize	
Reject low:	0.000000	
Reject high:	0.920000	

Nun im zweiten Durchgang sollten wir aber eine Normalisierung durchführen, da wir das Ergebnis für unser Endergebnis benötigen. Normalize soll also diesmal angewählt werden. Als Output diretory wählen wir einen neuen Ordner mit dem Namen Komet_abgezogen. Den Prozess CometAlignment lassen wir noch offen, denn den brauchen wir dann später noch ein letztes Mal. Wir können ihn aber minimieren.

Jetzt können wir mit Blink die neu erstellten Bilder durchschauen. Die Bilder sollten nun nur noch Sterne enthalten. Ein paar Pixel vom Kometen können aber noch übrig sein. Das liegt daran, dass die Rohbilder nicht alle gleich hell sind. Wenn es gar nicht klappt, weil die Bilder zu unterschiedlich sind, können wir versuchen beim CometAlignment LinearFit zu aktivieren. Das gleicht das Kometenbild an die Rohbildern etwas an. Es hat aber den Nachteil, dass dann das Ergebnis nicht mehr korrekt von der Helligkeit ist. Wir benötigen nämlich den Stack dieser Bilder wieder für eine weitere Bearbeitung.

Nachfolgend öffnen wir ImageIntegration und laden die neuen Bilder rein. Hier soll nun ein Bild entstehen, das nur Sterne enthält, aber keinen Kometen. Da auf den Einzelbildern der Rest des Kometen nun eher dunkel ist, müssen wir uns eher um Rejection low kümmern.

Lösen wir erst mal den Prozess mit den Standard-Einstellungen aus und sehen uns das Ergebnis an. Dann ziehen wir wieder ein Preview und wählen es als Region of Interest. Jetzt tasten wir uns langsam an niedrigere Werte bei Rejection low ran, bis wir zufrieden sind. Wenn eine minimale Spur des Kometen übrig bleibt, macht das nichts, denn diese wird dann später durch unser Kometenbild abgedeckt.



Das Ergebnis speichern wir dann unter "stars" ab.

6. Ergebnis von Ausgangsbildern abziehen

Als nächsten Schritt öffnen wir wieder unseren aktiven Prozess CometAlignment. Da nun das Bild "stars" von allen Rohbildern abgezogen werden soll, stellen wir unter Substract folgende Werte ein: Bei "Operand image" laden wir unser Bild "stars" rein und wählen bei "Operand is" Stars aligned an. LinearFit bleibt abgewählt und Normalize angewählt. Anschließend wählen wir ein neues Output Directory mit dem Namen "Sterne_abgezogen" und lösen den Prozess ein letztes Mal mit global apply aus.

Format Hints
Output
Output directory: 1/20190209Iwamoto1D/Bearbeitung2020/sterneabgezogen1 🗎
Prefix: Postfix: Overwrite
Parameters 🏠
466.119508 Y 454.438425 Show _M3G8306_c_cc_d_20200
642.464380 Y 323.596125 Show _M3G8569_c_cc_d_20200
dX 39.472831 dY -29.287588
Subtract
Operand image: 190209Iwamoto1D/Bearbeitung2020/masters/ <mark>stars.xisf</mark> 🗎 🕙
Operand Origin: Operand is: Drizzle save: ImageIintegration Stars aligned ImageIintegration DrizzleIntegration Comet aligned ImageIintegration
Enable LinearFit Normalize
Reject low: 0.000000
Reject high: 0.920000
Interpolation ±
Pixel interpolation: Bicubic Spline
Clamping threshold: 0.30

7. Bilder integrieren, verbessertes Kometenbild wird erstellt

Als Nächstes müssen wir die entstandenen Bilder integrieren. Als Ergebnis sollten wir nun ein Bild erhalten, das in sauberer Form nur noch den Kometen enthält. Dieses sollten wir abspeichern.



Auf dieses Bild können wir noch ein ABE, oder DBE anwenden, um die Gradienten in diesem Bild zu reduzieren. Als Correction sollten wir Substraction wählen und Normalize anwählen, damit sich das Bild farblich nicht ändert. Dadurch können wir später das Bild anhand der Sterne farbkalibrieren. Dieses Bild benennen wir um in "comet2".

8. Bilder zusammenfügen

Als letzten Schritt müssen wir jetzt nur noch die Bilder "comet2" und "stars" zusammenfügen. Wir öffnen also beide Bilder und den Prozess Pixelmath. Nun könnten wir beide Bilder ganz einfach addieren, allerdings würde sich dadurch die Helligkeit ändern. Damit das nicht passiert, ziehen wir wieder den Median des Bildes ab, das wir hinzu addieren. Wir bringen also das Bild "stars" in den Vordergrund und schreiben folgende Zeile in

Pixelmath: \$T+comet2-med(comet2) . Jetzt ziehen wir das Dreieckchen auf das Stern-Bild und der Komet sollte nun im Bild erscheinen.



Das kommt der wahren Helligkeit des Kometen am nächsten. Je nach Geschmack kann man aber den Kometen auch noch mehr betonen. Wir machen den letzten Schritt wieder rückgängig und erweitern die Formel in Pixelmath etwas: \$T+(1.5*comet2-med(comet2)). Damit erscheint der Komet 1,5 mal stärker.

Der Rest der Bearbeitung unterscheidet sich nicht von jedem andern Deep Sky Bild. Da Kometen oft sehr viel Dynamik enthalten (heller Kern, schwacher Schweif) kann man versuchen mit HDR Multiscale und/oder etwas nachschärfen Details aus dem Kometenkern herauszuarbeiten. Natürlich ist es von Vorteil, wenn man hier das Bild vor der Vereinigung benutzt.

Wenn man das Bild anhand der Sterne farbkalibrieren will, sollte man aber das Kometenbild vor der Vereinigung weder strecken noch farblich verändern.

Viel Spaß bei der Bearbeitung, Tobias Lindemann

Falls Fragen auftauchen, könnt ihr mich gerne unter folgender Adresse kontaktieren: TobiasLindemann@iss-tracking.de