

HSCデータ解析講習

古澤 久徳 2022.11.28

HSCのデータ解析に関する情報

すばる共同利用ユーザ データ解析ヘルプデスク
helpdesk@hsc-software.mtk.nao.ac.jp

- ハワイ観測所公式 HSC装置ページ
 - <http://www.naoj.org/Observing/Instruments/HSC/>
 - ・ HSCについて不明な場合はまず参照しましょう。Helpdeskへのリンクもあります。
- ハワイ観測所公式 HSCデータ解析helpdeskページ
 - <http://hsc.mtk.nao.ac.jp/pipedoc/>
 - ・ パイプライン配布所へのリンクやFAQがあります。
- SSPで管理しているデータ解析・DB情報ページ
 - <https://hsc-release.mtk.nao.ac.jp> (SSP一般公開データ)
 - <https://hscdata.mtk.nao.ac.jp> (SSP内部向け公開データ)
 - ・ 日本人研究者・学生はCollになれます → 質問はhelpdeskまで

HSCとは

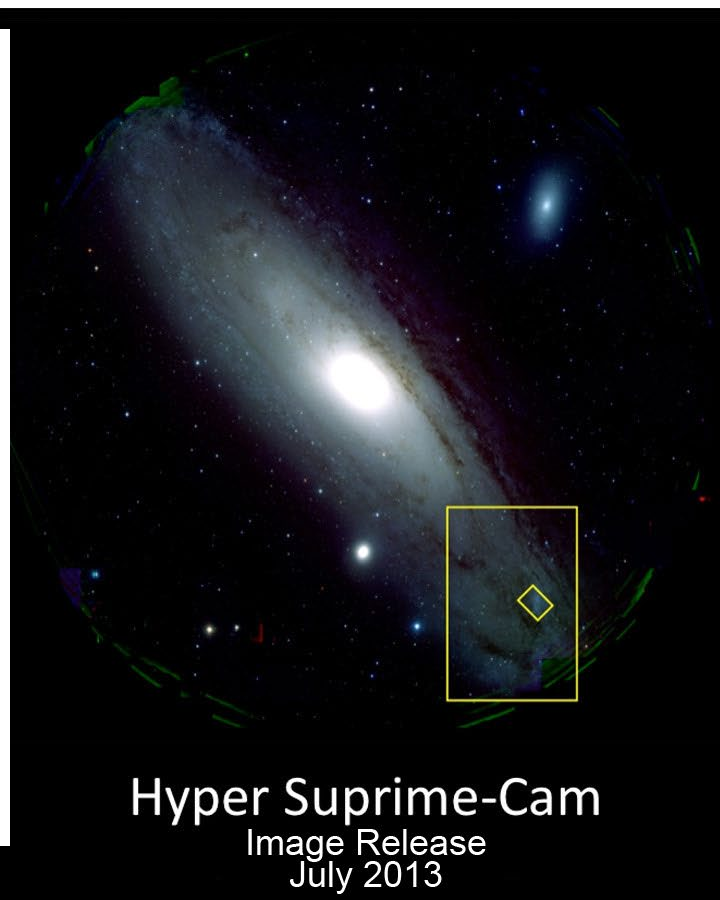
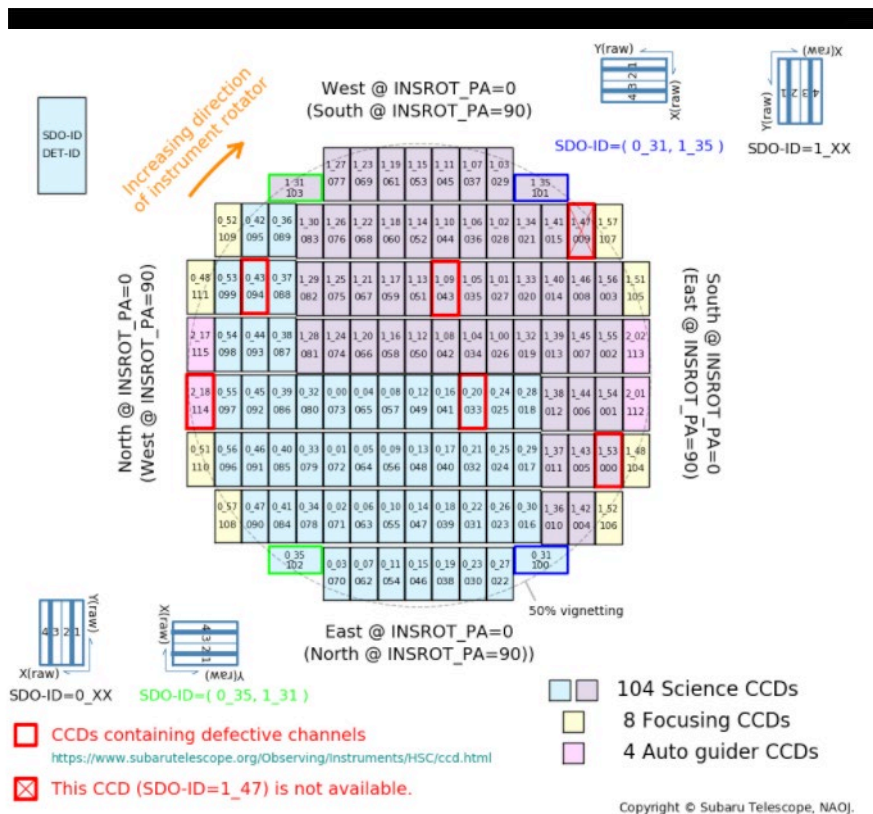
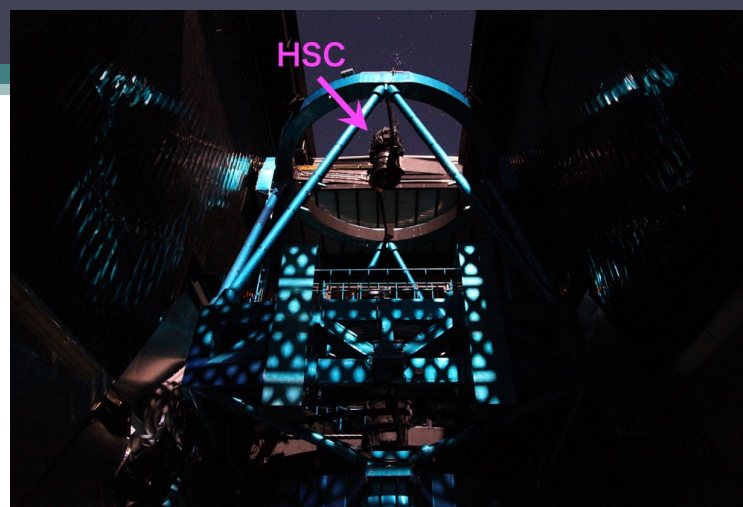


Image Release
September 2001

Hyper Suprime-Cam
Image Release
July 2013

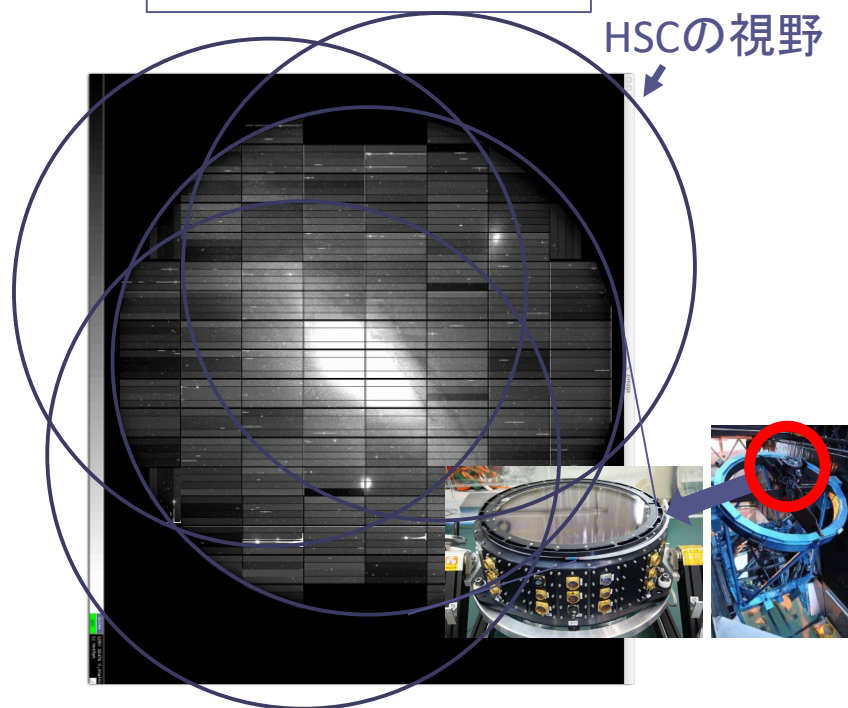
解析の目的と流れ

● 目的

- CCDごとの生データから諸々の邪魔な混入要素を取り除き、
- 必要に応じてCCDデータを足し合わせてS/Nや広さを増やし、
- 天体からのphotonのエネルギー密度とその分布を測定すること
- ただし、実際の天文学ではバンドシステムを定義し、そのバンド内で観測されるエネルギー密度を等級として求める

モザイクCCDカメラによる
ディザリング観測

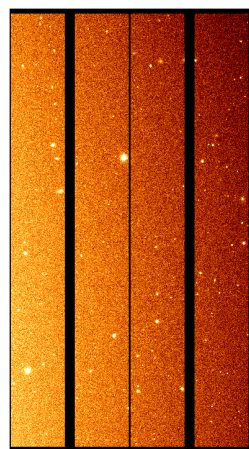
104 CCD x 感度差
25 % vignetting
3% distortion



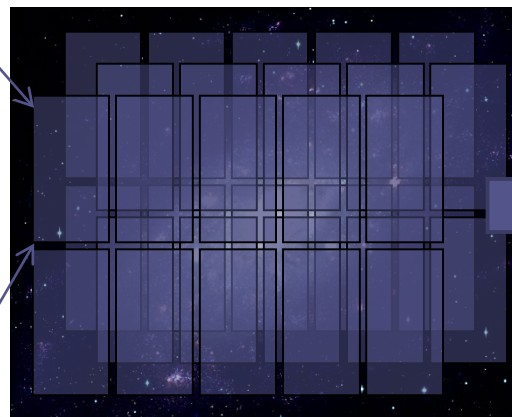
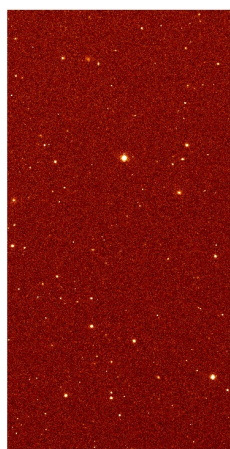
一般的な撮像データ解析手順

1. 各CCDのバイアス除去・感度差補正（一次処理）
2. 必要な一次処理済みCCDデータを集め、
3. モザイクして足し合わせる
4. 位置情報、カウントを物理量に直す

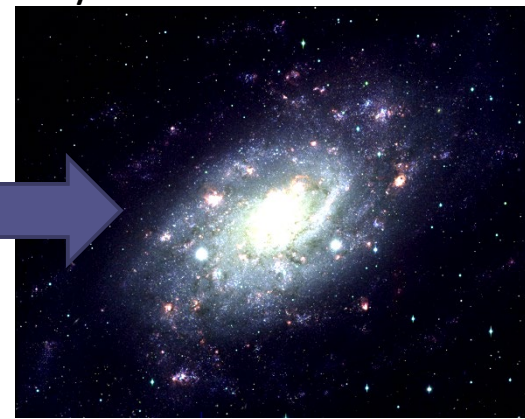
HSCデータ解析
→ これらをカバー



個々のCCD



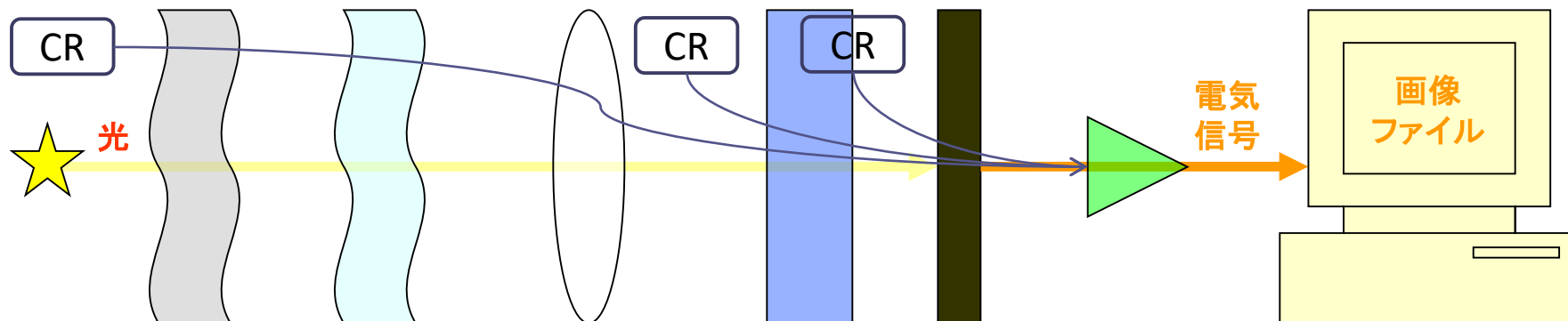
ディザリングして
取得した複数ショット



スタック画像
これを用いて天体カタ
ログを作成

解析で補正する混入要素

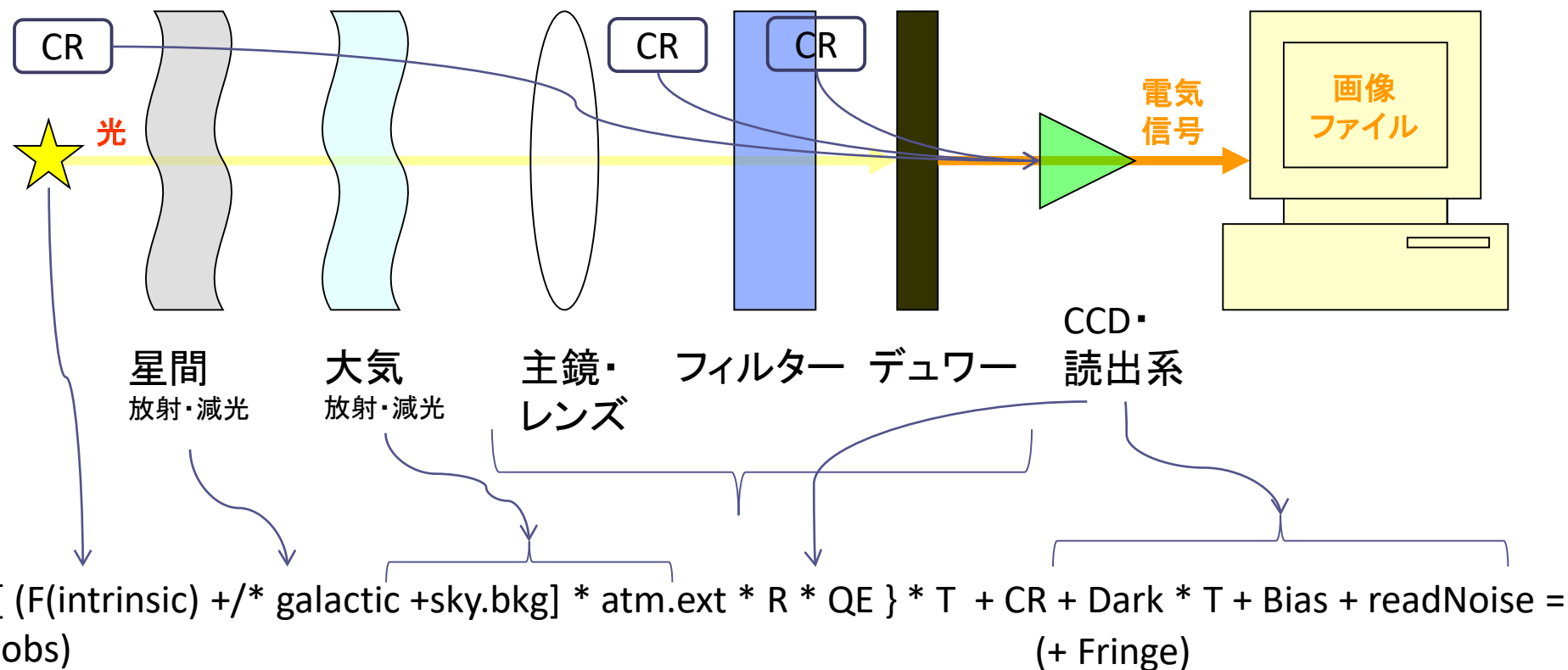
- 基本的には右から左にむかって補正していく



星間物質	大気	望遠鏡光学系	フィルター	CCD	読出エレキ
吸収・散乱	吸収・散乱 Seeing Sky	吸収・反射 Geom.Distortion	吸収・反射	QE Bad pixels Dark Fringe	バイアス クロストーク
文献・色で補正	標準星 Smoothing/Modeling Sky sub.	Flatfielding Dist.Correction	Flatfielding	Flatfielding Triming Dark sub. Fringe sub.	Bias sub. Crosstalk corr.

解析で補正する混入要素

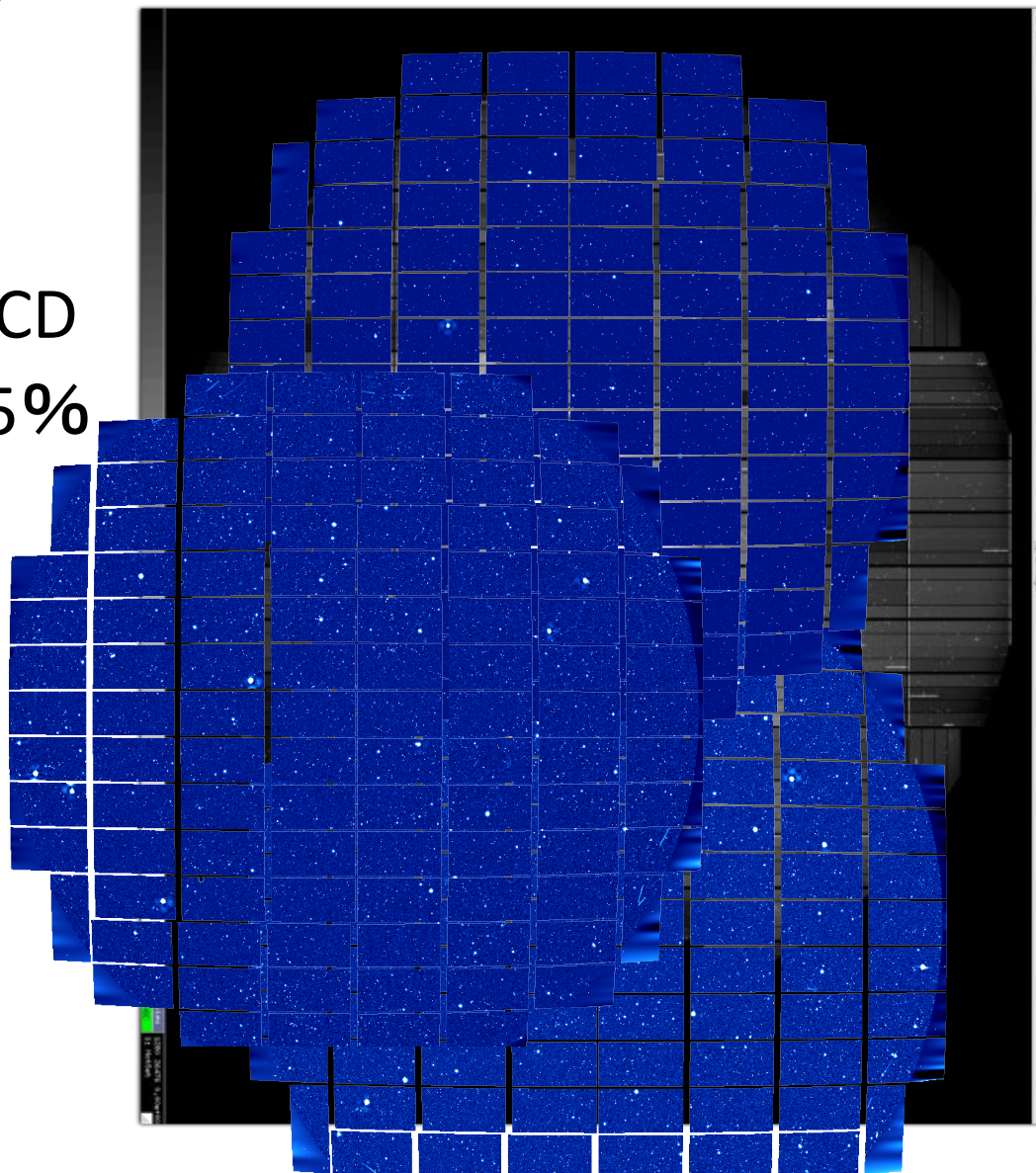
- 基本的には右から左にむかって補正していく



HSCデータの特徴

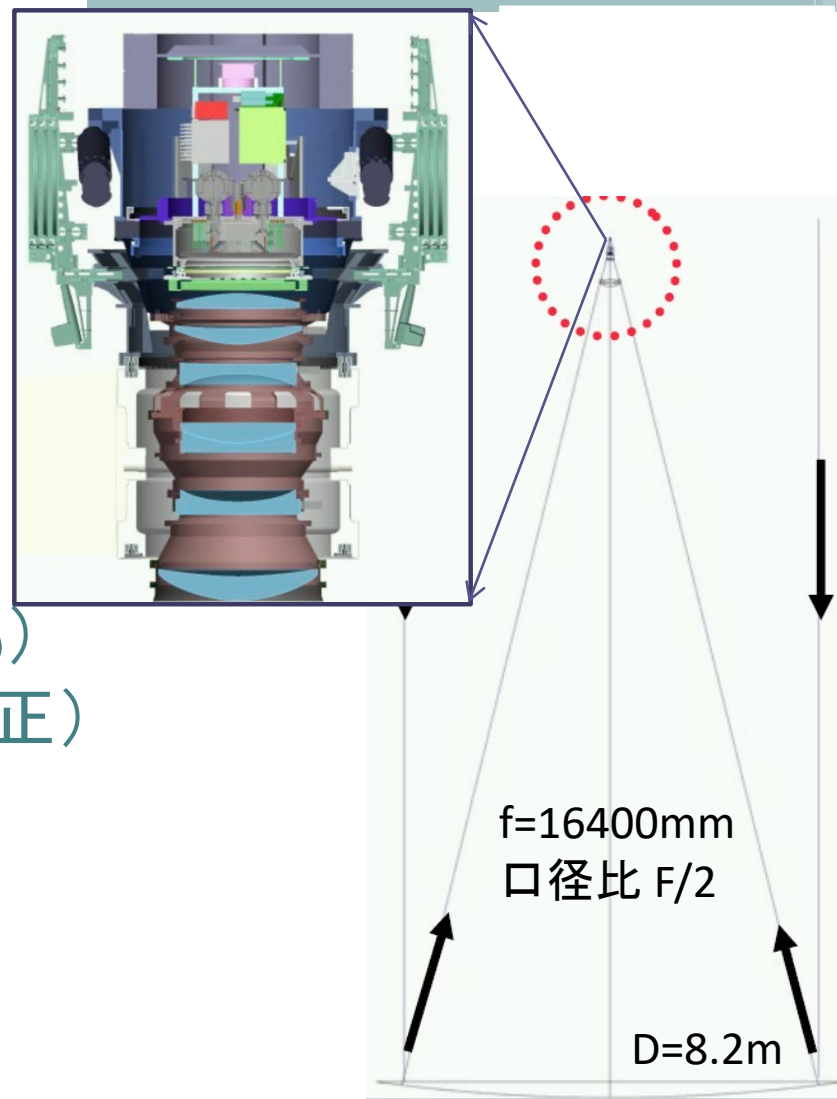
- 1.5度視野
= 感度差のある
104個の科学画像CCD
- 光学けられ視野端で25%
- 光学歪み視野端で3%

モザイクCCDカメラによるディザリング観測



望遠鏡と装置

- 望遠鏡の駆動
 - 目的方向で日周運動を追う
 - ・ 温度、風速／向、ドーム制御
 - 主鏡制御
 - カメラの姿勢制御(焦点方向も)
 - レンズ制御・ADC(大気分散補正)
- 装置の駆動
 - CCDでのデータ取得
 - ・ CCD、フィルター、シャッター
 - 観測制御系へのデータ転送



Derived from
Miyazaki-san's slide

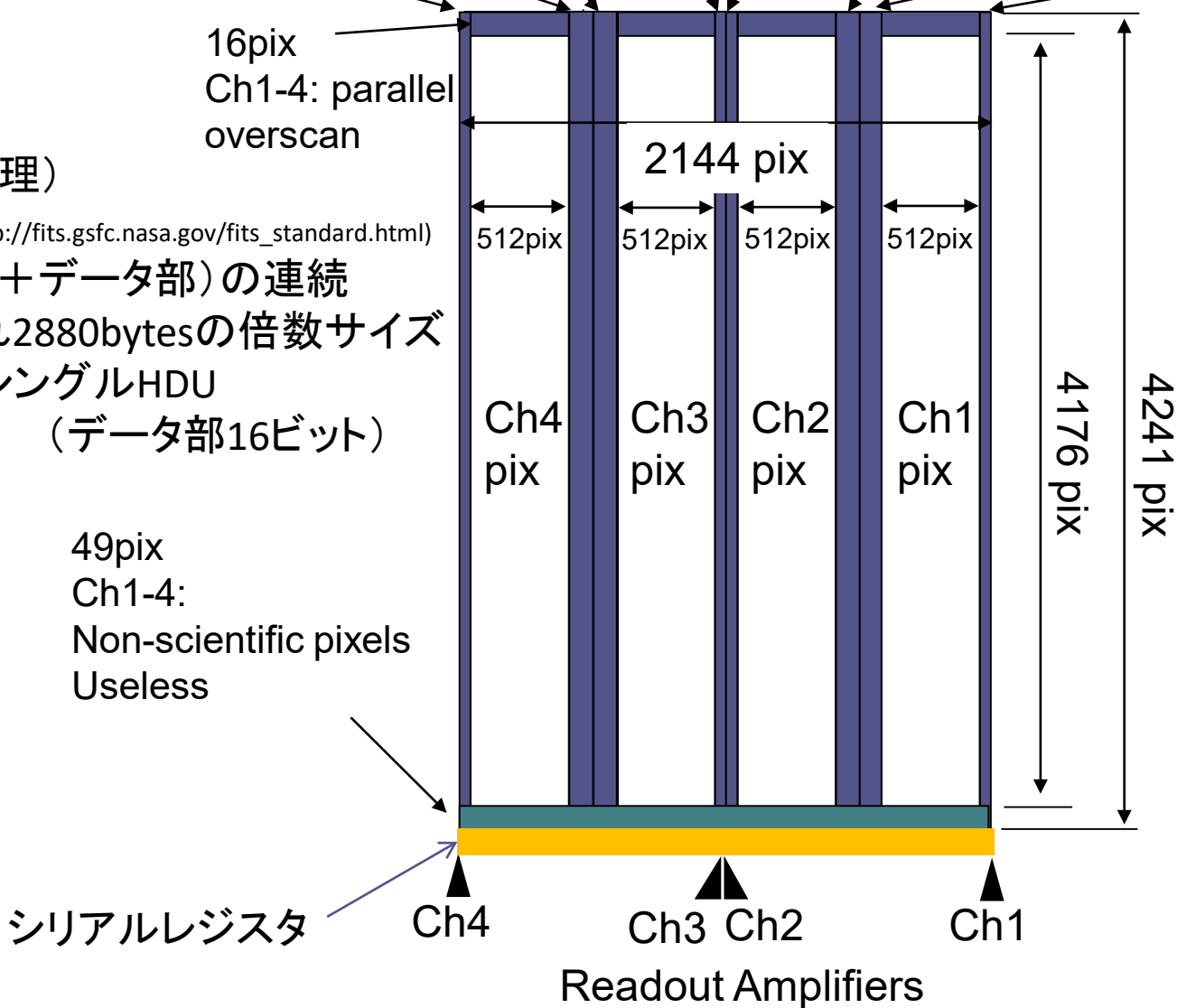
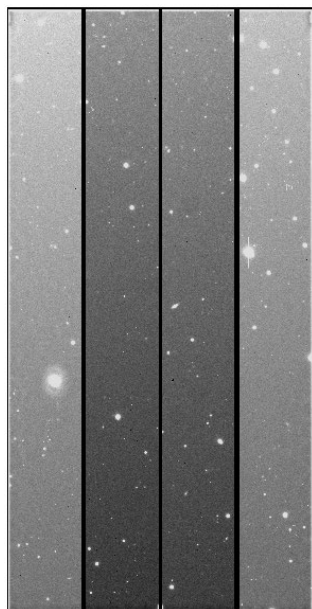
HSC CCDの 生データ(未処理) フォーマット

8pix 16pix 16pix 8pix 8pix 16pix 16pix 8pix
 Ch4: Ch4:serial Ch3:serial Ch3: Ch2: Ch2:serial Ch1:serial Ch1:
 blank **overscan** **overscan** blank blank **overscan** **overscan** blank
 領域 領域 領域 領域

HSC生データ(未処理)

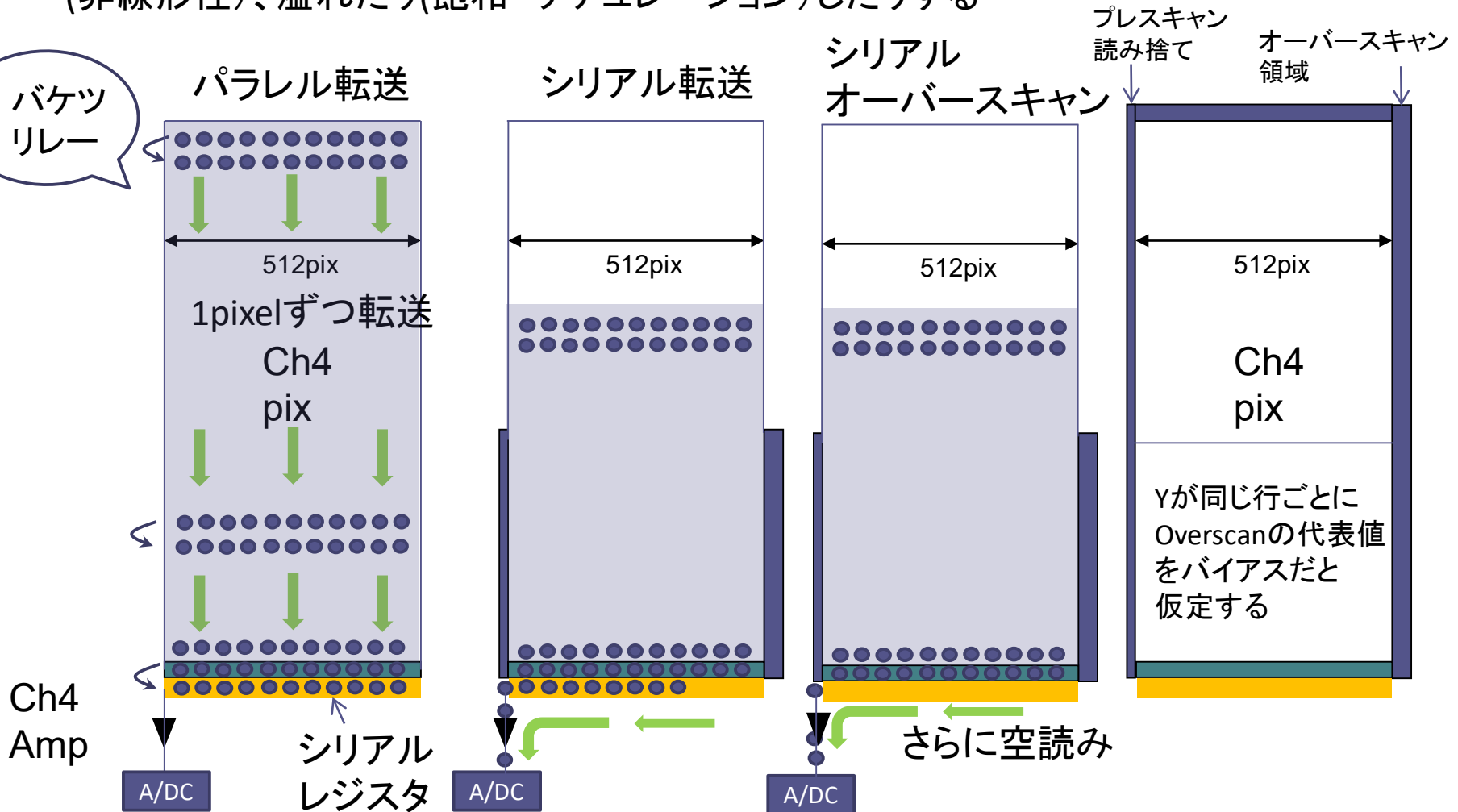
FITSフォーマット (http://fits.gsfc.nasa.gov/fits_standard.html)

- HDU=ヘッダ部(+データ部)の連続
 - 各部分はそれぞれ2880bytesの倍数サイズ
- 1 CCDごとに1 FITSシングルHDU
 (データ部16ビット)



HSC CCDの読み出し

- CCD (Charge Coupled Device) は光電効果によって生じた電荷を電圧として読み出す
- その電圧を増幅AD変換(全画素1画素ずつ通る)したカウント(ADU)を画像として保存する
- 応答 (電圧->読み出し後カウント) の線形性は高いが、電荷が溜まりすぎるとずれたり (非線形性)、溢れたり(飽和・サチュレーション)したりする



生画像名のルール: HSCA<8桁の数字>.fits . 8桁の数字はvisitとCCD (SOD-ID)の組み合わせからなる

例1) visit=22348 , SOD-ID=0_04 (DET-ID=065) => HSCA02234804.fits

例2) visit=22348 , SOD-ID=1_04 (DET-ID=034) => HSCA02234904.fits

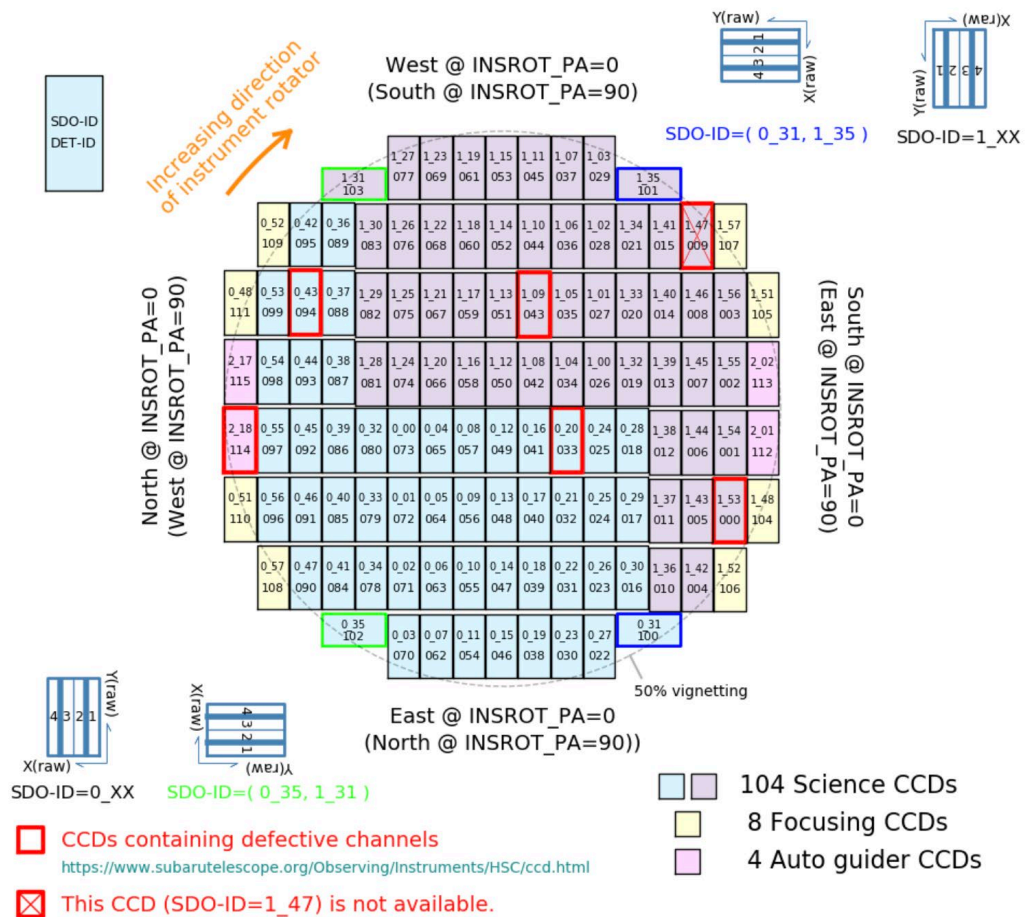
Hyper Suprime-Cam CCD arrangement

(updated in August 2021)

Visit :

HSC1ショットのこと。1Visit = 112 CCDのデータからなる。生画像名ルールとも関連するが、visit IDは偶数しかない。

CCD ID : 2種類あるので注意。生データに使用されるSOD-ID (例: 1_04) とhscPipe解析時の識別番号であるDET-ID (例: 034)。



データの呼び名 FRAMEID, EXP-ID, visit, ccd

- STARS内はCCDごとに管理されていて本名はFRAMEID (HSCA%08d)
 - 1ショット(112CCD)は HSCA00000000 – HSCA00000057, HSCA00000100 – HSCA00000157 で表現される。
- 1積分112CCDを1個のIDで参照しようというのがEXP-IDとvisit
 - hscPipeのVisit は1積分内先頭のFRAMEIDの下2桁を落とし、%6d に丸めた値 (HSCA*のうち。HSCB系に進むと%7dになる予定)
 - 例
 - FRAMEID: HSCA12345600 – HSCA12345757 → visit=123456
- 各CCDのFITSにDET-IDというキーワードが割り当てられている
 - hscPipeのccdはDET-IDと同じ値を採用

FITSファイル画像Viewer

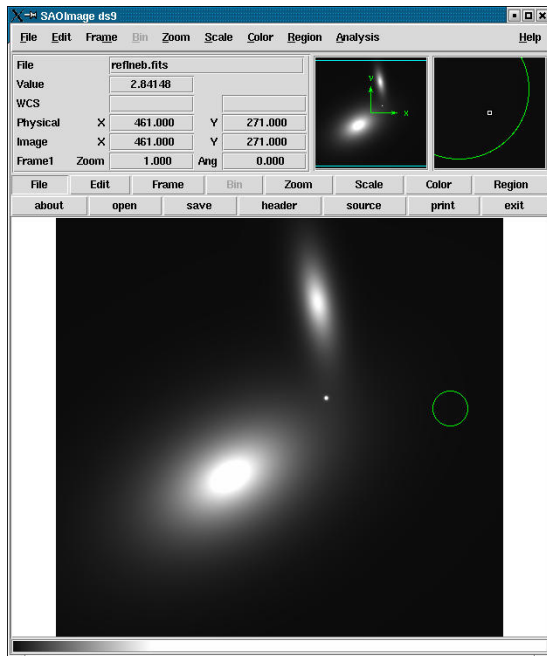
- ds9

- 開発: SAO

- <http://hea-www.harvard.edu/RD/ds9/>

- 特長

- 画像操作がしやすい
 - WCSを使って天球座標でタイルしたり
 - XPAなどで外部プロセスから操作できる



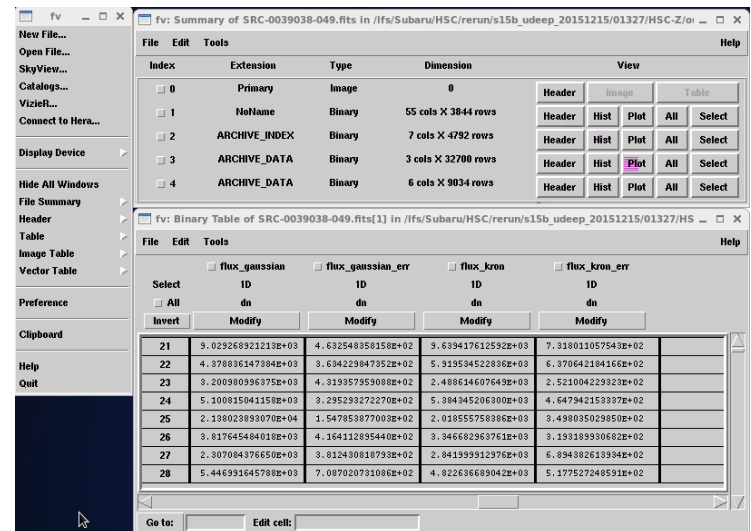
- fv

- 開発: NASA HEASARC

- <http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/ftools/fv/>

- 特長:

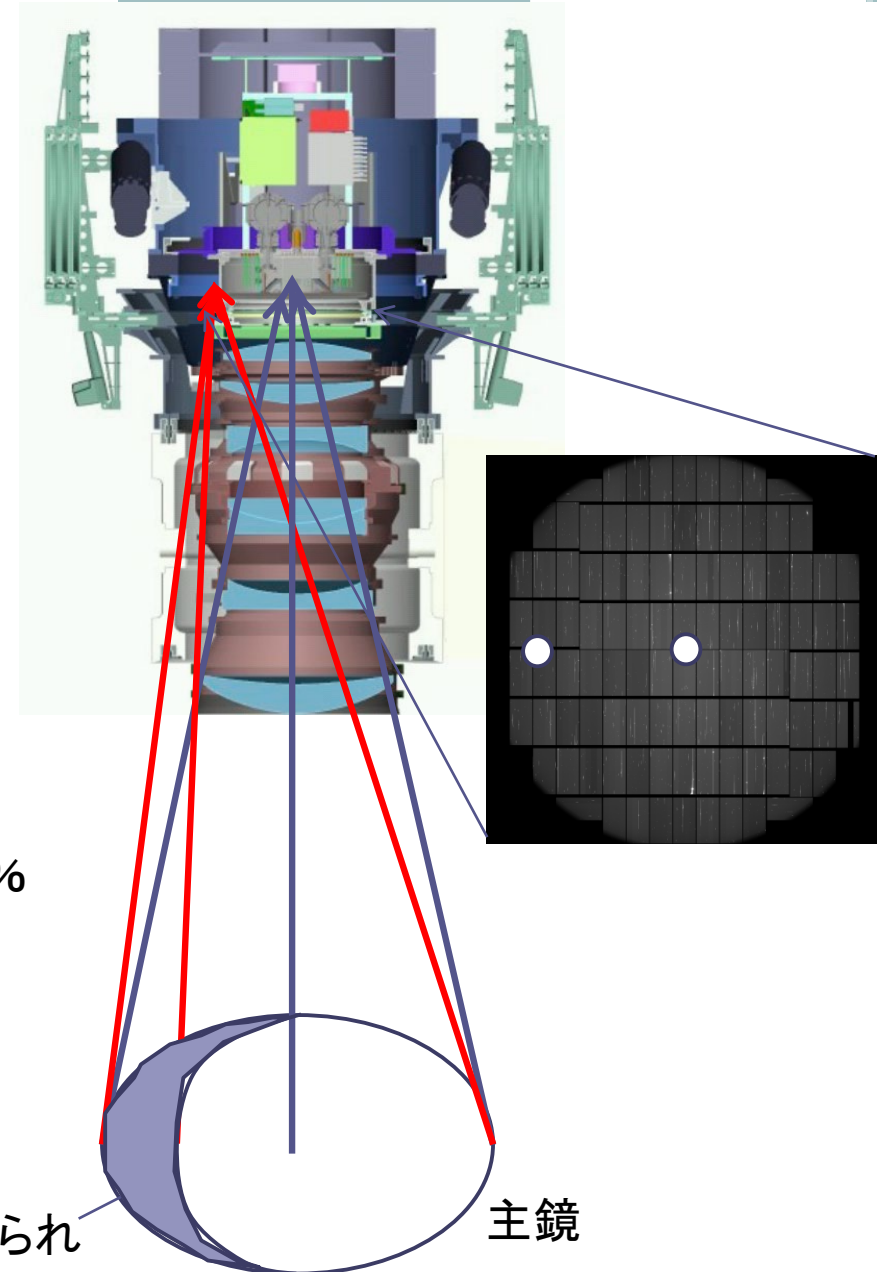
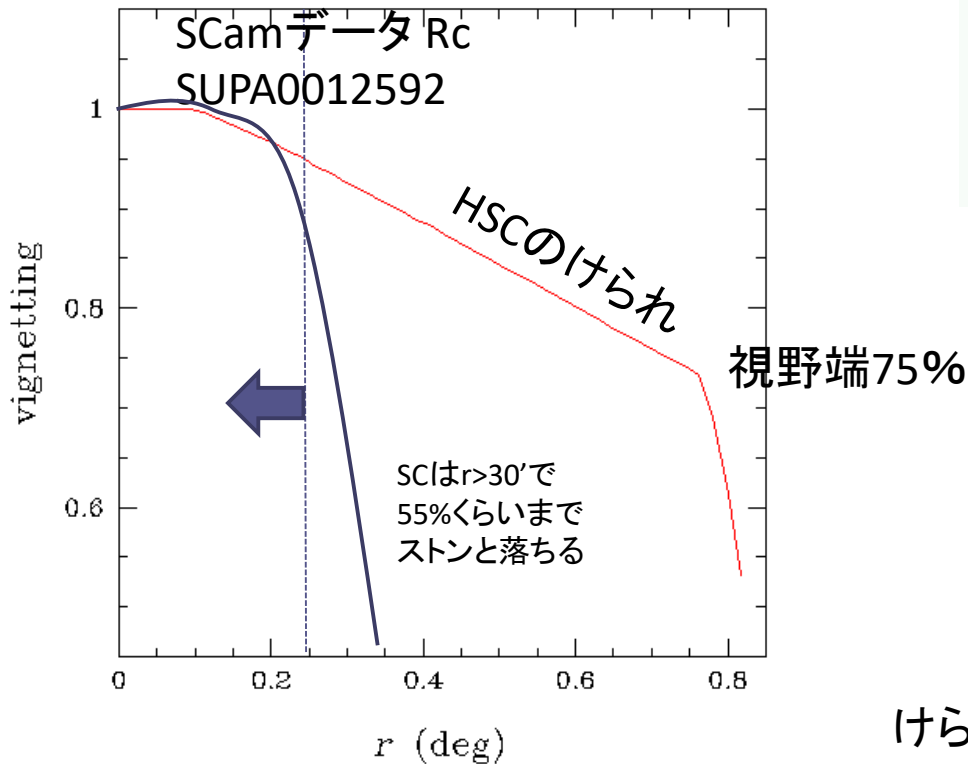
- 複数HDUやBINTABLE形式を操作しやすい



他にも便利なツールはあります

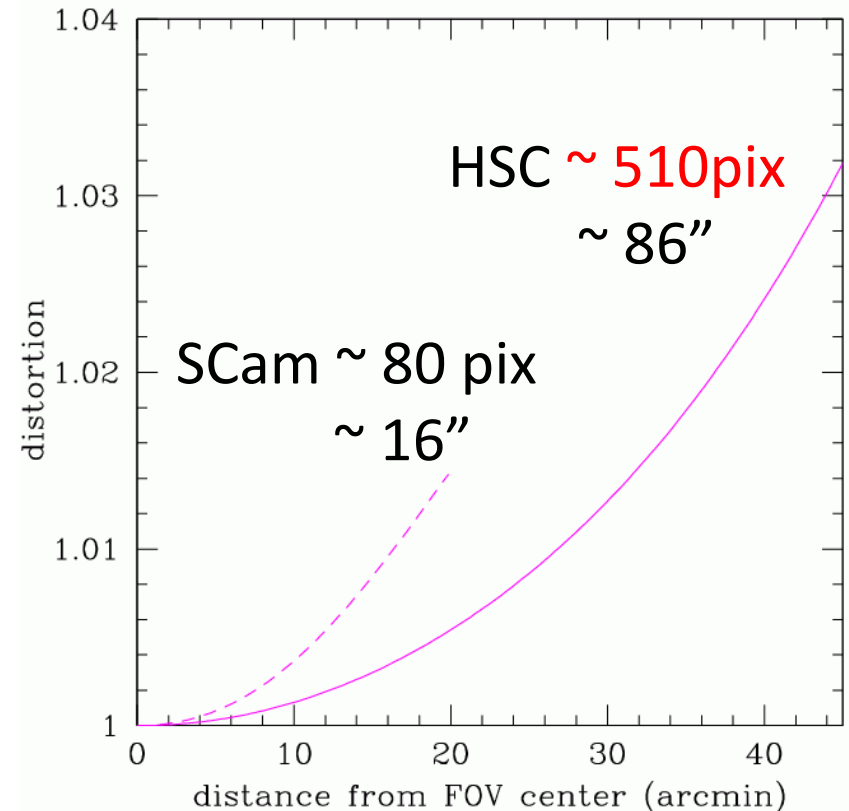
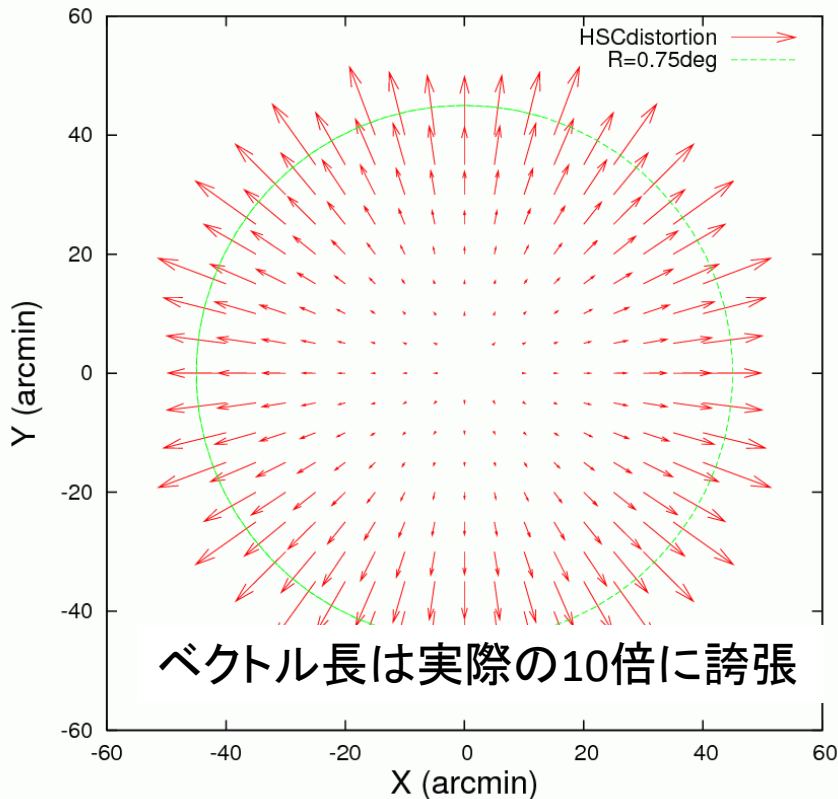
光学系けられ

- 視野中央はけられない
- 視野端にいくほどけられる



光学系ディストーション

- 収差の一つで、位置ズレをおこす。視野端で3%
- その微分量として、1 pixが見込む空の面積は中心より視野端で7%くらい小さい



HSCデータ解析ソフトウェア hscPipe

- HSCすばる戦略枠観測 (SSP) 用にNAOJ・IPMU・プリンストン大で開発したもの
 - LSST用パイプラインをHSC-SSPに最適化したもの
 - 最近はLSSTのパイプラインに集約された。使い勝手がこなれるにはもう少し時間がかかるのでこの講習ではhscPipe 8.5.3を使う
- Pythonによりパイプライン動作を記述
- あまり混んでいないフィールドについては一般共同利用観測データにも使うことができる
- 広がった天体・混んだ天体・SDSS／PanSTARRSでカバーされないフィールドは非対応と考えて良い
 - 共同利用ユーザの需要を鑑み、徐々に対処策を検討したい

hscPipeのパッケージ群

- ソフトウェア構成
 - 様々なモジュールの組み合わせからなる
 - GNUのツールや無償ソフトウェアを組み合わせて構成されている
 - LSSTの解析に向けて開発されており、HSCほか、様々なプロジェクトのデータ解析できるよう、パイプラインの基盤となるモジュールに加えて、各プロジェクト(装置)専用のモジュールやタスクを追加することで解析を実現している。
- 言語
 - パイプラインの枠組はPythonで書かれている
 - 画像処理や高速な手続きはC++、pybind11でPythonから利用できるようにしている

ソフトウェアパッケージの成り立ち

- EUPSというソフトで用いるソフトのバージョンコントロールをする
 - `setup hscPipe 8.5.3`
 - `setup -j -r` ローカルにあるパッケージ名 でタスクのオーバーライド
- `src`
 - C++コード
- `include`
 - C++ includeファイル
- `lib`
 - C++のライブラリ: C++でリンク & Pythonコードから呼ぶ
- `python`
 - パイプラインのモジュール
- `ups`
 - パイプラインを実行するための環境変数を設定
 - EUPSの`setup`コマンドにより読み込み

生データ・CALIBデータの管理

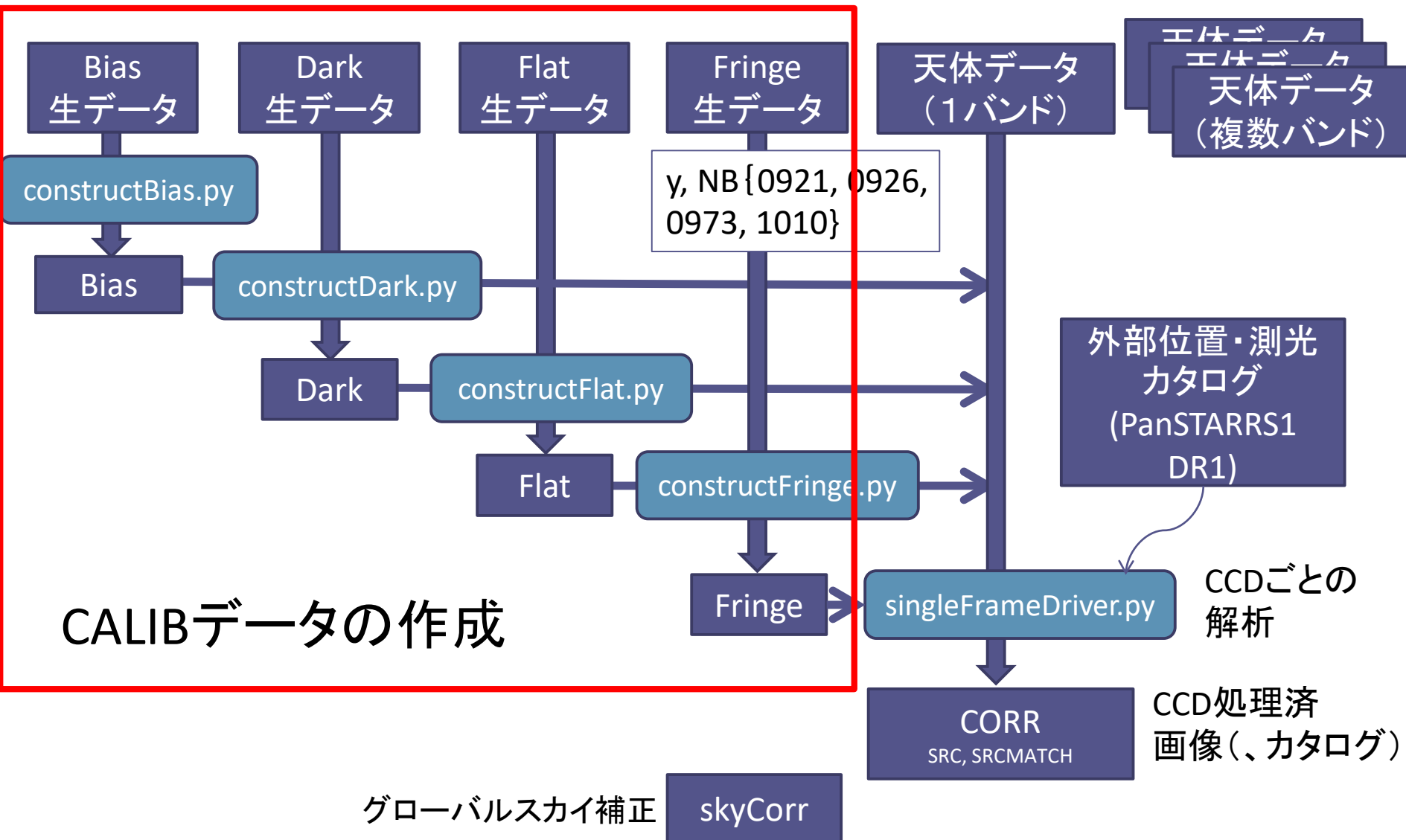
- 生データとCALIB (BIAS, DARK, FLAT, FRINGE,SKY)は、専用のディレクトリツリー下に置かれ、sqlite3のデータベースで管理される
 - registry → 生データ
 - calibRegistry → CALIBデータ
 - 各CALIBデータには有効期間が設定されている
- 次のCCD解析のプロセスの中では、解析データのDATE-OBSとフィルターに応じて、適用期間や条件を満たすCALIBデータがデータベース問い合わせで選択され、読み込まれる

Calibrationデータ

- 観測所のHSC装置ページに使うべきFRAMEIDの範囲がリストアップされているので確認すること
- https://www.naoj.org/Observing/Instruments/HSC/calib_data.html
- あと、どのデータをどのように扱うかで悩んだら、helpdesk+観測所、データ取得に明るい人と確認 & 議論すると良い
- アーカイブにそのような有用な情報を不足なく載せるのが課題

A. CCD解析

解析の流れ(1): CCDごとの解析

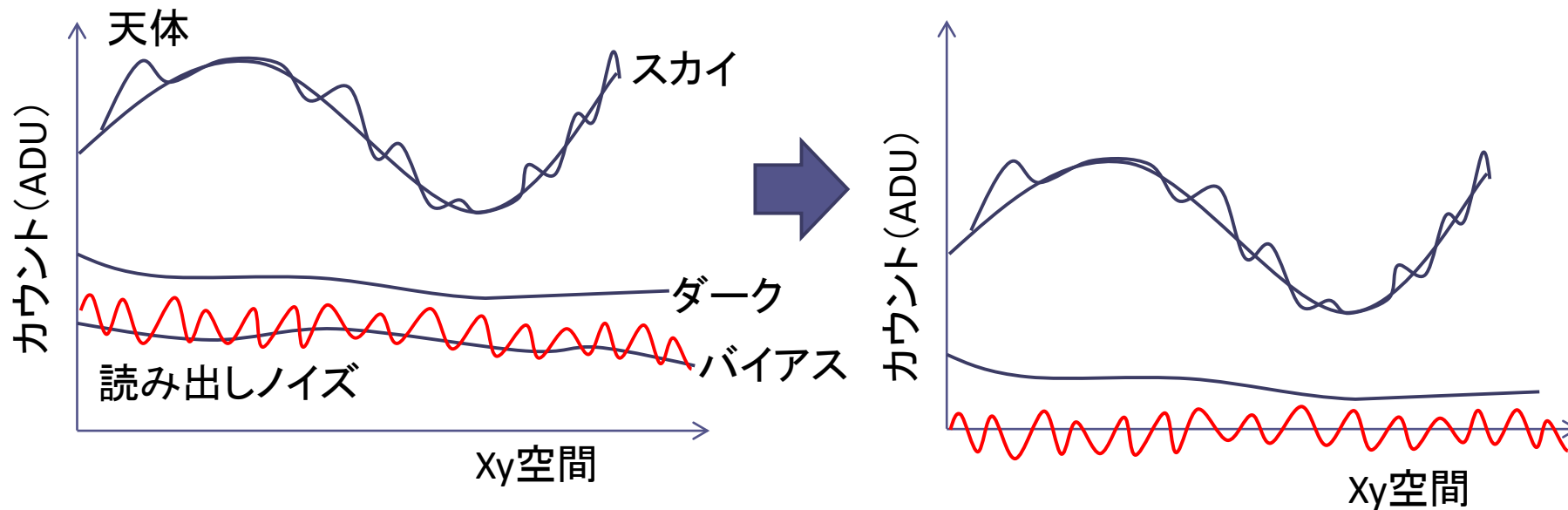


1. バイアス・ダーク作成(適宜スキップ)

- 利用データ
 - 毎観測ラン1度、解析用にBIAS $x \sim 10$ 、DARK (300s) $x \sim 5$ を取得している
→ 不安ならばhelpdesk、Support Astronomer に確認
 - 毎夕・朝のテストデータはヘルスチェック目的なので解析には不適
- バイアス作成
 - 手続き： オーバースキャンを引き・トリムし、メディアンスタック
 - 入力： バイアス $x \sim 5-10$
 - 出力： CCDごとにシングルHDUのfloat32 FITS画像
- ダーク作成
 - 手続き： オーバースキャンを引き、バイアスを引き、メディアンスタック
 - 入力： ダーク $x \sim 3-5$ 、(バイアス)
 - 出力： CCDごとにシングルHDUのfloat32 FITS画像
- メモ
 - HSCの場合、ダークは暗電流補正というよりは迷光除去の目的もある
 - オブジェクトデータの解析でもオーバースキャンは引かれる。バイアス・ダークフレーム作成・それらの引き算は適宜スキップしても、数%以上の精度ではさほど問題ないだろう
 - むしろ良質のバイアス・ダークを取得することが課題

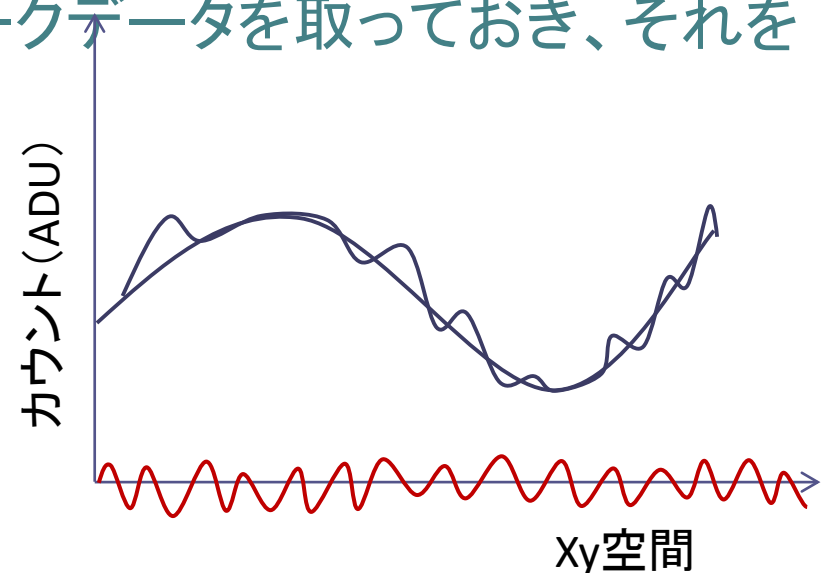
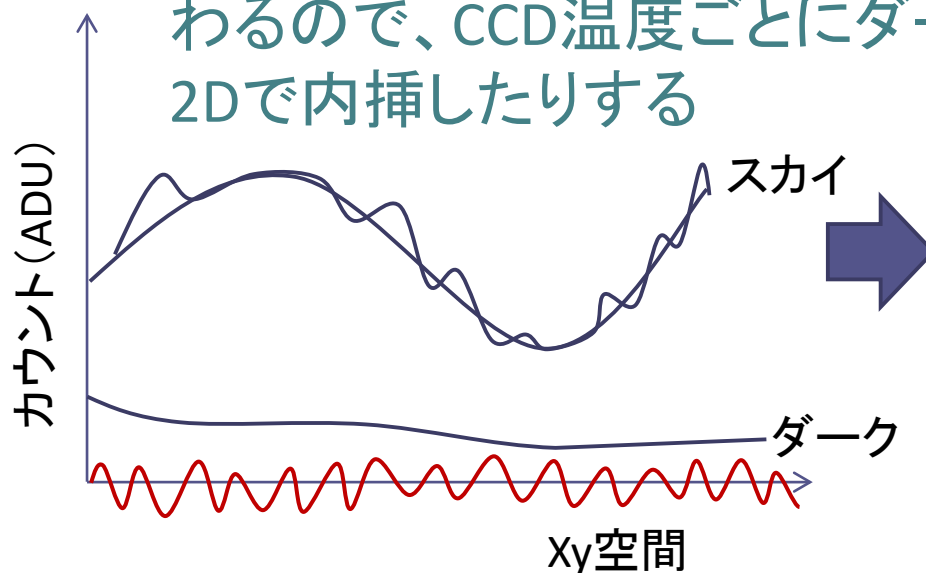
1. バイアス引き

- 天体の光や積分時間に関係なく読み出し時に加えられる電圧 (=カウント) の下駄を2Dで引く
 - AD変換される入力電圧が常に正となるように加えられる
- 変動成分をオーバースキャンで引き、残差成分を2Dで引く



2. ダーク引き

- 時間とともにわき出る暗電流によるカウントを2Dで引く
- 一般に低温($\sim < -100^{\circ}\text{C}$)で制御されていれば十分小さい
- SSP: 時間に比例すると仮定し、300秒ダークのカウントをスケールして引く
 - 本当にダークが効くようなケースでは2Dパターンが温度で変わるので、CCD温度ごとにダークデータを取っておき、それを2Dで内挿したりする

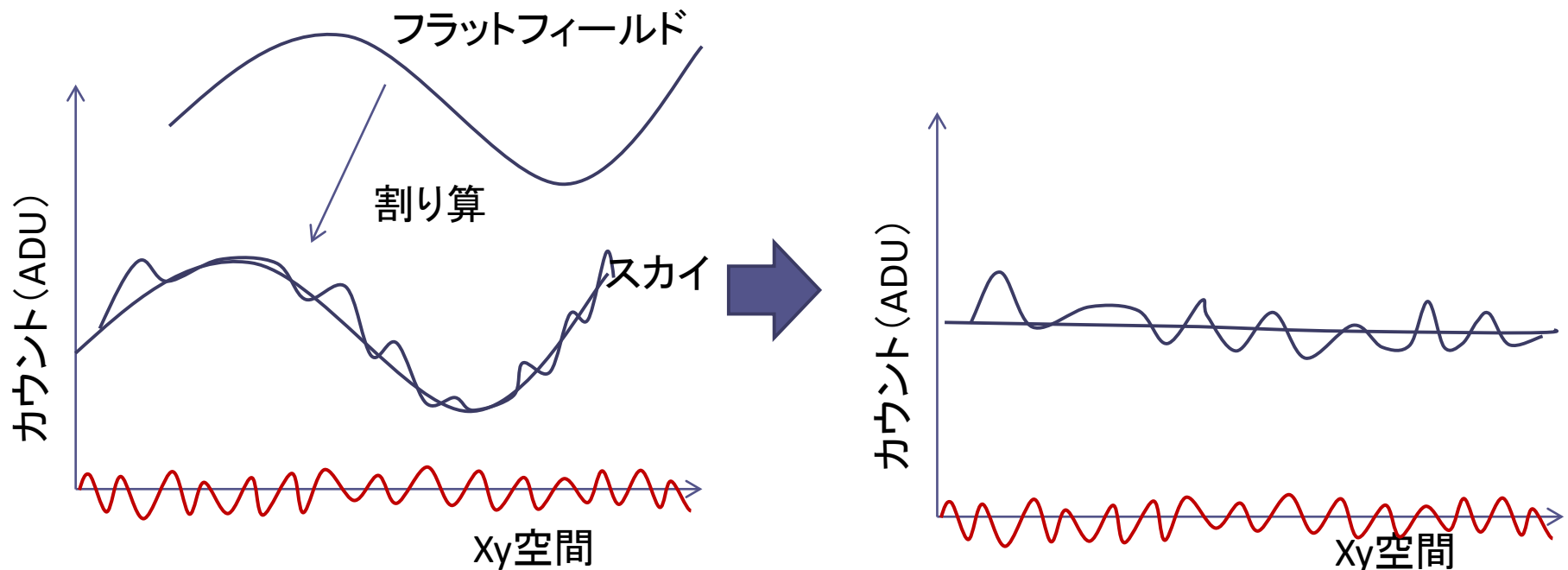


2. フラット作成

- 利用データ
 - ドームフラット
 - ・ 観測ランごとに各バンド10ショット程度。規定ランプ設定・電流安定下で取得したものを使う
- 手続き
 - 各CCD: オーバースキャン引き・バイアス引き・ダーク引き
 - 全体: 視野全体の平均値が1になるよう各CCDを規格化し、メディアスタック
- 入力
 - ドームフラット x~10 (+バイアス、ダーク)
- 出力
 - 各CCDごとに3 HDUのfloat32 FITS画像
- メモ
 - フラットフィールドのCCD間の相対的な感度・ゲイン比・vignetting比は保存される
 - フラットフィールドの視野中心は~1.5カウント程度になる

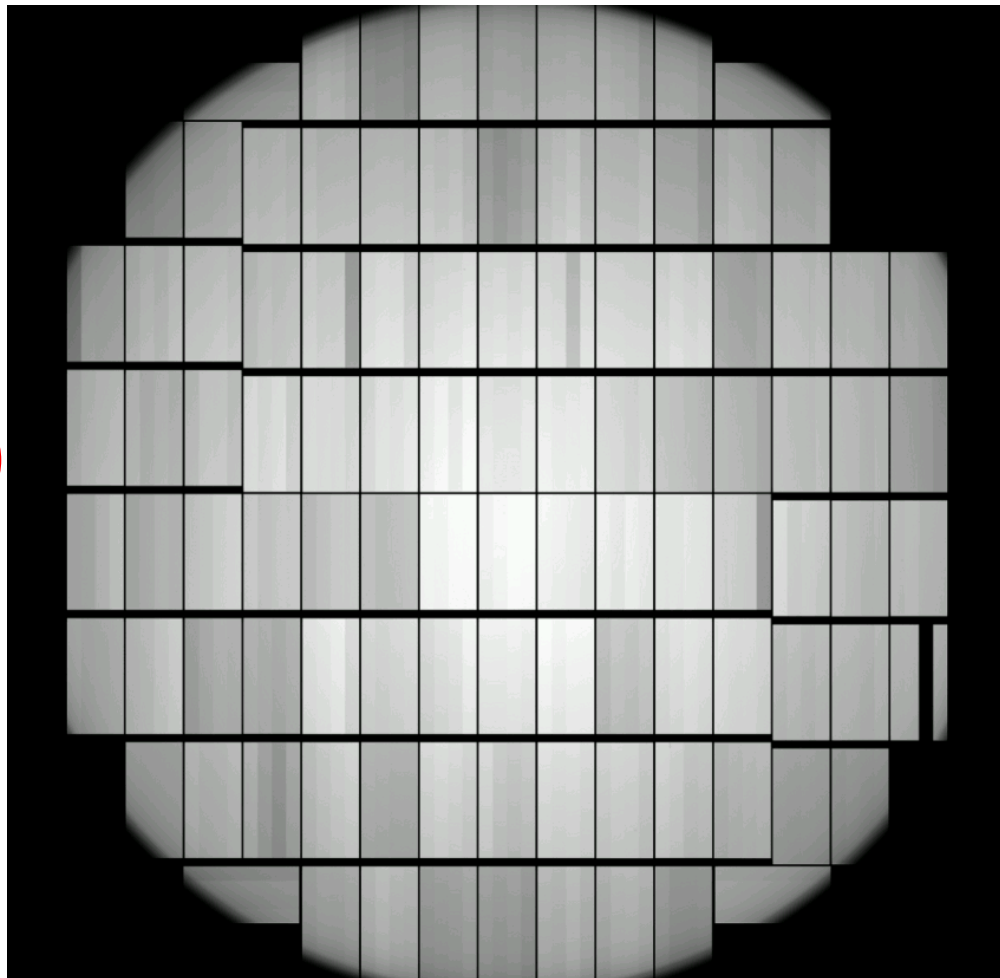
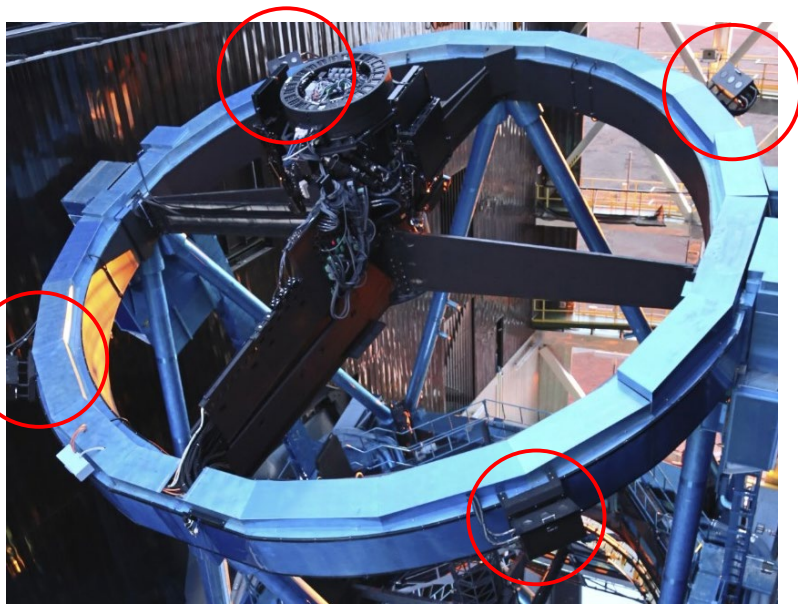
3. フラットフィールドディンク

- 望遠鏡から検出器 (CCD) までの正味の感度により入射カウントよりも減ってしまった分を2Dで割って戻す
- フラット (フィールド) と呼ばれる一様光源を取った画像を入射カウントに対する減光パターンの答えと仮定する

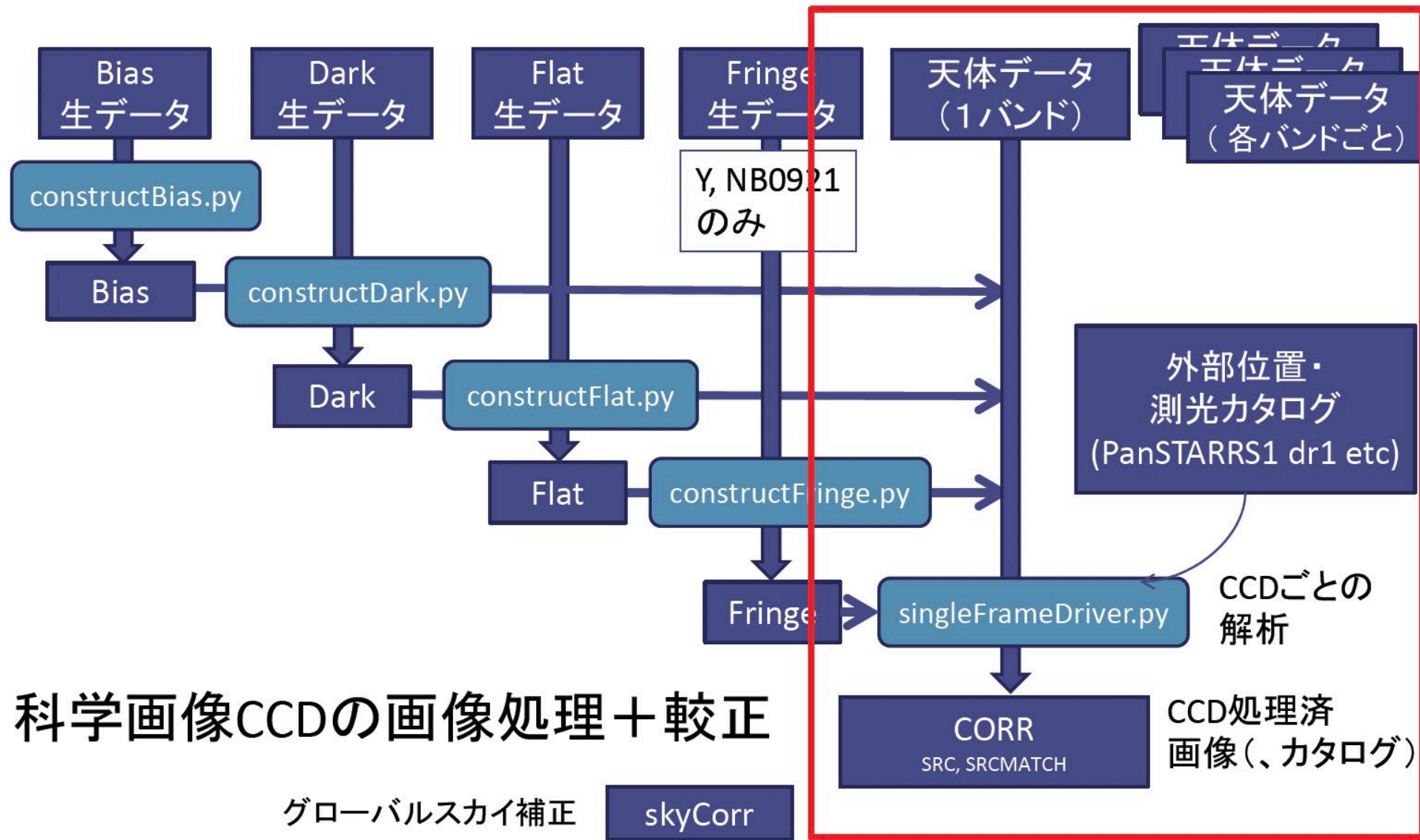


フラットフィールド画像

- HSC-12バンドの →
ドームフラット



解析の流れ(1): CCDごとの解析



解析手続きの非常に端折った解説

- CCDごとの解析 (reductionとも、較正含む)
 - 一般的に1次解析・Reductionと呼ばれる作業
 1. バイアスを引く
 2. 悪いピクセルを潰す (CR除去含む)
 3. フラットフィールドして感度を補正する
 4. 画像の歪みを画像上で修正、またはマッピングの決定
 5. スカイバックグラウンド (背景光) を引く
 - 一般的に2次解析・較正 (Calibration) と呼ばれる作業
 6. PSFを測定して、以後いろいろな測定に使う (1次解析の用途も含む)
 7. ピクセル座標を天球座標に変換するマッピングを行う
 8. カウント (ADU) から等級に換算する原点を決める
 9. 天体測定

4. CCD解析: singleFrameDriver.py

- 入力
 - ◻ オブジェクトデータ各ショット、CALIBデータ、外部カタログ
- 出力
 - ◻ 解析済み画像、カタログ類
- 手続き
 - ◻ 4-1. CCD解析パート
 - フラットフィールドディング(前処理もろもろ含めて)
 - ◻ 4-2. Calibrateパート
 - スカイ引き <- この段階では128pix mesh local sky引き
 - PSF(Point Spread Function)測定
 - アストロメトリ(天球座標較正)
 - 等級原点較正
 - ◻ 4-3. 天体測定パート
 - 天体検出・測定

アストロメトリ

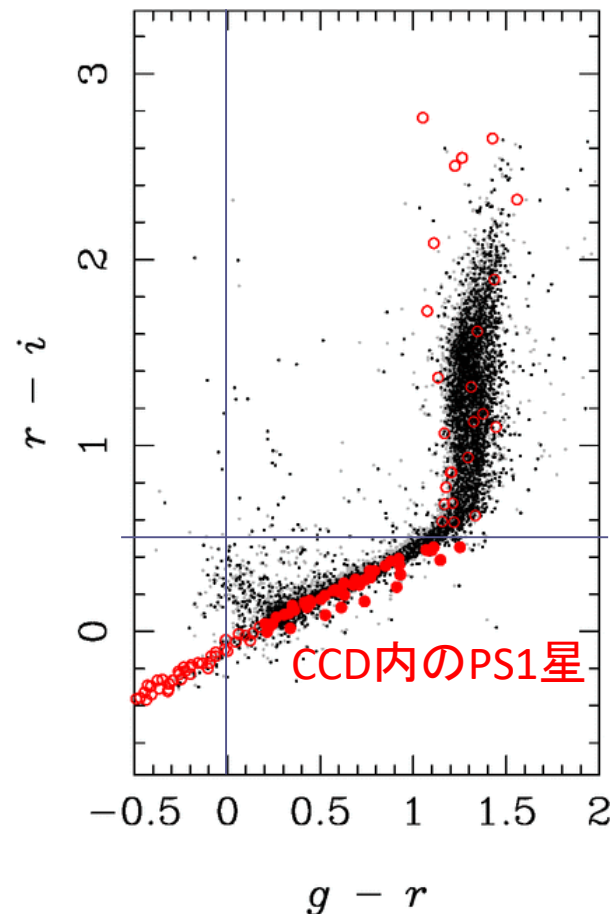
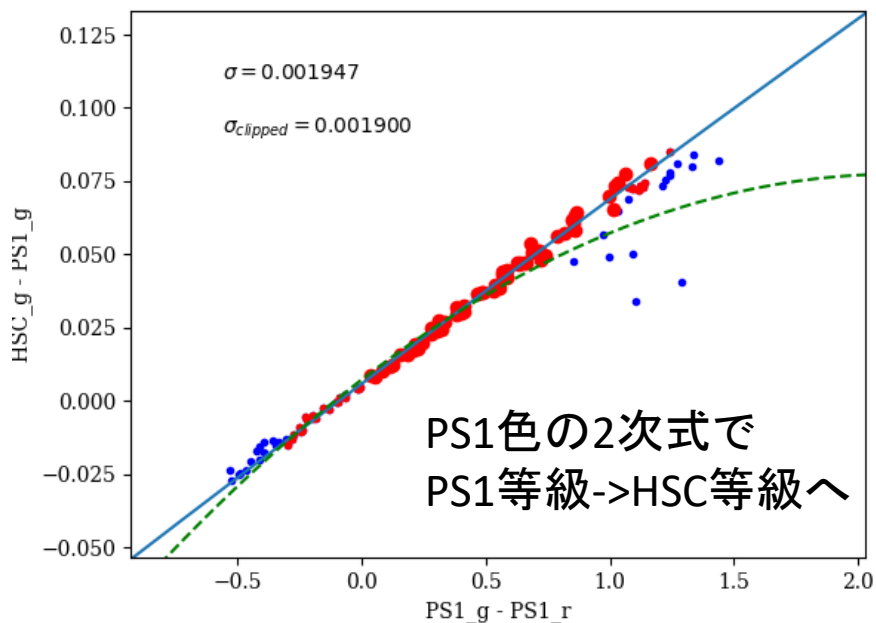
- ピクセル座標から天球座標 (RA, Dec) へのマッピング
- 1) 較正用の外部カタログとデータに写る天体とを同定・ペアリングする**クロスマッチング**と、2) $(Ra, Dec) = f(x, y)$ の関係を決める**WCS決定**の2段階からなる
- マッチング
 - 較正用の座標カタログはPanSTARRS 1 DR1
 - Tabur+2007 → Pessimistic Pattern Matcher (PPMb)に変更
 - 150個以下のピンホールパターン、混んだ領域に強い
- WCS決定
 - TAN-SIP (3次) の係数決定 (→ 出力のヘッダへ)
 - 後方のHDUにASTによるマッピング結果も保持
- メモ
 - これらはCCDごとに行っているため、写りの良し悪し・カタログ天体の多少の影響を受けることがある

等級原点較正

- カウント (Instrumental Flux) を Flux (=HSC等級) に直すためのスケールング因子 (オフセット) を決める
 - $m[\text{mag}] = -2.5 * \log_{10}(\text{Instrumental Flux}[\text{ADU}]/\text{FLUXMAG0})$
 - $= -2.5 * \log_{10}(\text{count} [\text{DN}]) + a$ (等級原点)
- 手順 (Instrumental magnitude)
 1. 外部カタログ (PanSTARRS1) と CCD に写る天体とを **クロスマッチング**
 - $g-r > 0$ 、 $r-i < 0.5$
 2. カatalog天体の HSC等級と CCD 上のカウントの関係
 $m = f(\text{count})$ (上の a) を決める **等級原点決め**
 - 恒星状 (点源) 天体、 $r=12\text{px}(\sim\phi 4'')$ で測った **フラックス** を使う
 - → Aperture Correction の必要性
 - 結果の FLUXMAG0 は、以前は先頭ヘッダにあったが、後ろの方の HDU に移動しかつデータ部に格納され、陽に見えなくなった

等級原点較正

- HSC native band等級で較正
 - 参照等級はPS1->HSCの変換
- 較正に使う星の色範囲
 - $g-r > 0$ 、 $r-i < 0.5$



4-3. 天体検出測定部

- 検出

- スムージング

- PSFでスムージングして天体のピークを増幅する

- ピーク検出

- 局所的なノイズより十分大きいカウントを持つ
 - N-pixel以上繋がったところを探す

- 重なり合ったピークを持つ天体のフラックスの分離計算(デブレンド)

- 測定

- 位置(ピーク・重心座標 x, y)

- 形状(2次モーメント、楕円の方位角、銀河モデル化)

- 明るさ

- 開口(アパーチャ)測光

- 固定開口 : circular aperture flux

- 可変開口 : Kron flux

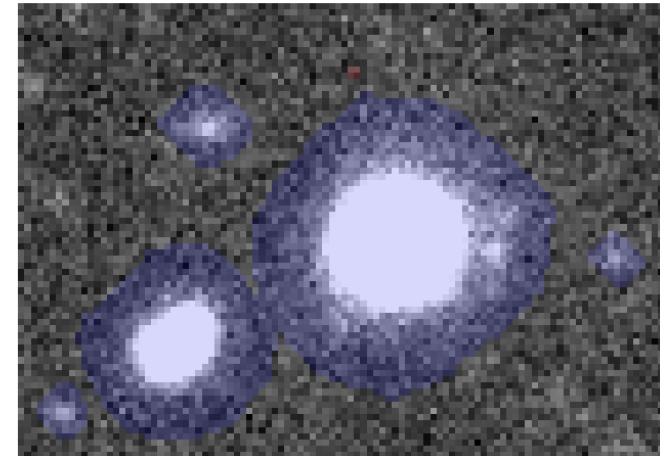
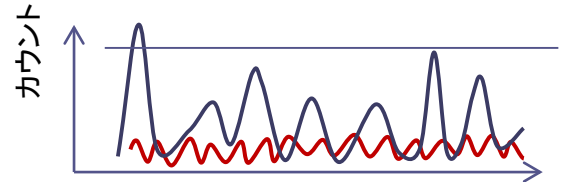
- モデル測光

- PSF flux

- CMoDel flux – Composite Model flux = 銀河を2成分モデルでフィット

- フラッグ

- 天体測定したピクセルのマスク情報や測定中のイベントにより割当

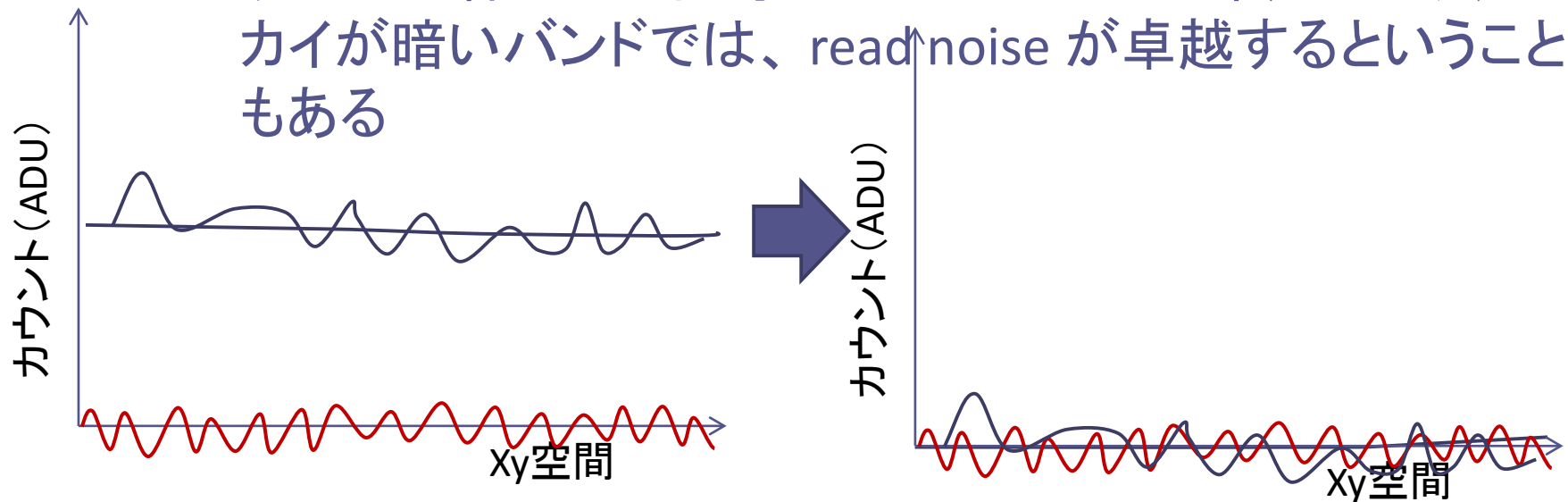


-0.044 -0.02 0.005 0.029 0.054 0.078 0.1 0.13 0.15

footprint

5. スカイ(バックグラウンド)引き

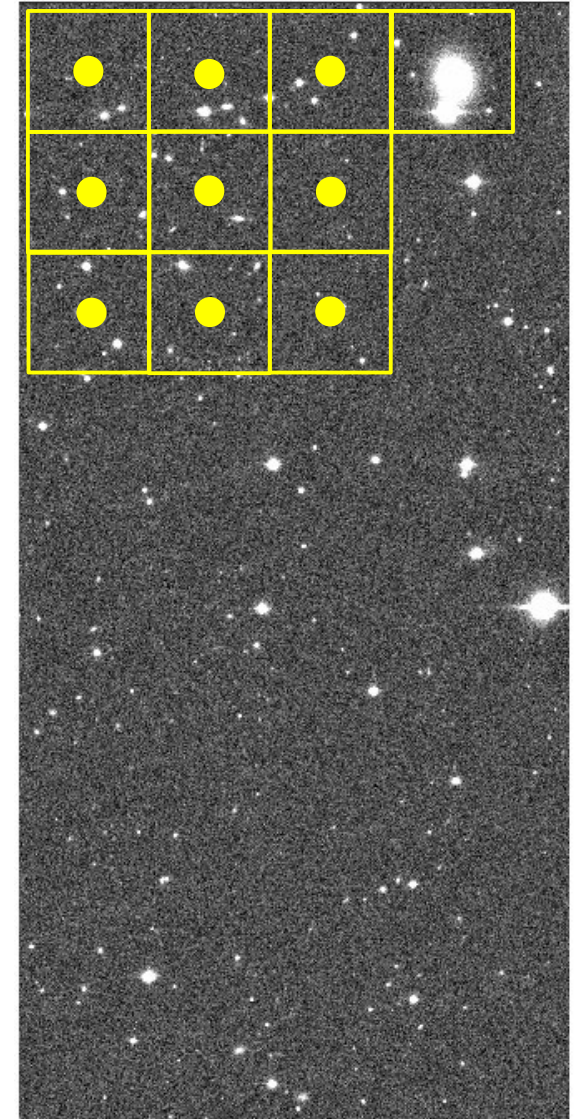
- 夜空は明るいので、CCDに結像するカウントの大部分はスカイバックグラウンド(背景光)
 - ピクセルごとのスカイのポアソンノイズが、暗い天体に対しては測定時のノイズとして卓越する
 - 明るい天体はFlux自身のPoisson noiseが卓越したり、スカイが暗いバンドでは、read noise が卓越するということもある



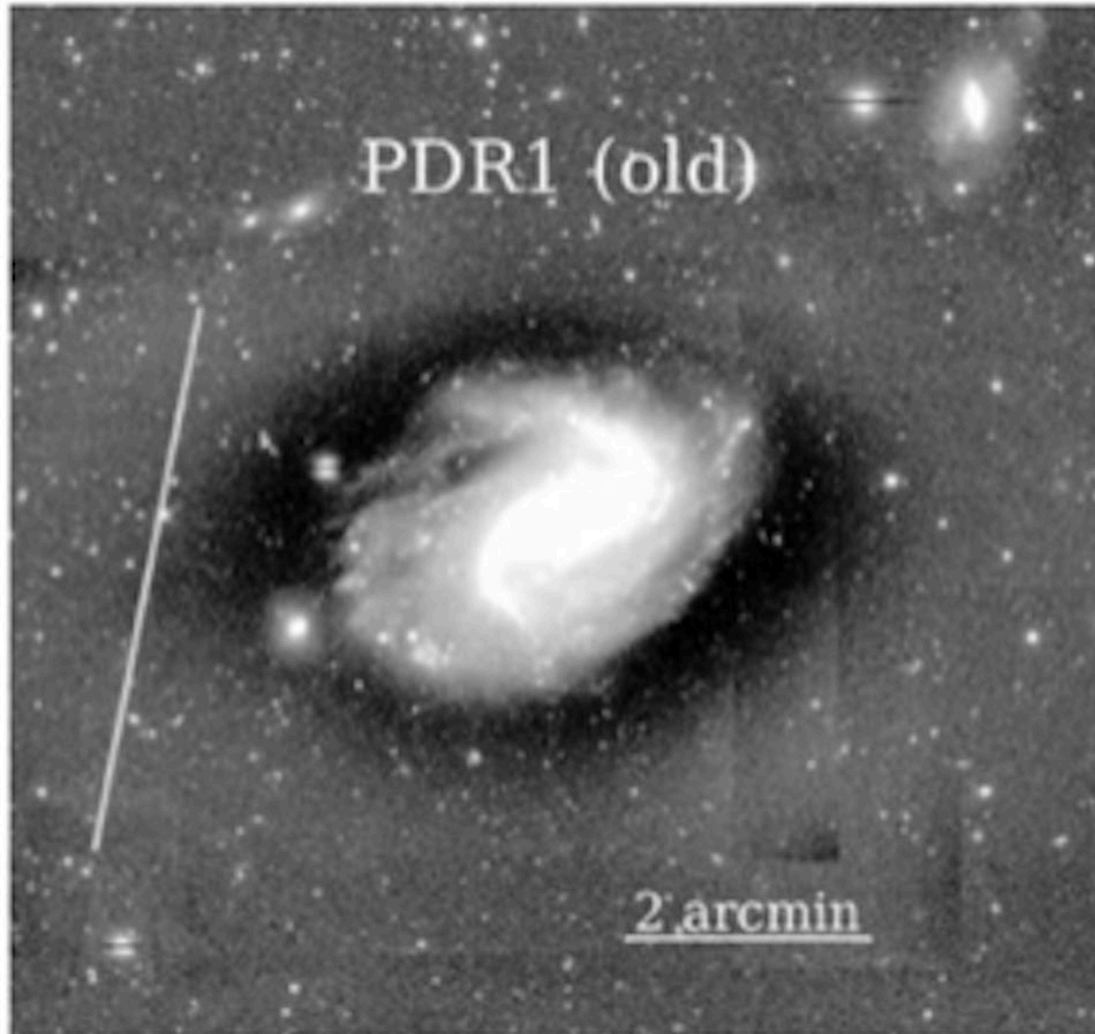
5. スカイ引き(2)

- 一般論
 - ◻ スカイを引いて、カウントのゼロレベルを決めない限り、天体測定は出来ないので、実は測定にクリティカルな作業。その割にそう単純ではない。
- よくある方法
 - ◻ 対象天体が大きい・領域自動決定がむずかしい場合
 - 天体外の場所を指定し、内挿で面フィットして引く
 - ◻ 対象天体が小さく、ほどよくブランクな領域
 - スカイ引きしたい領域を矩形などのメッシュに区切り、その中のモードなど、天体からのカウントの影響を受けづらい代表値を求める。ソフトによってはサブメッシュずらしながら行う
 - その代表値をスムージング(ex. medianフィルター)したり、関数フィットしたりして内挿して引く

一般論としてはそうだが、HSCの場合は難しさもあって、もう少し複雑



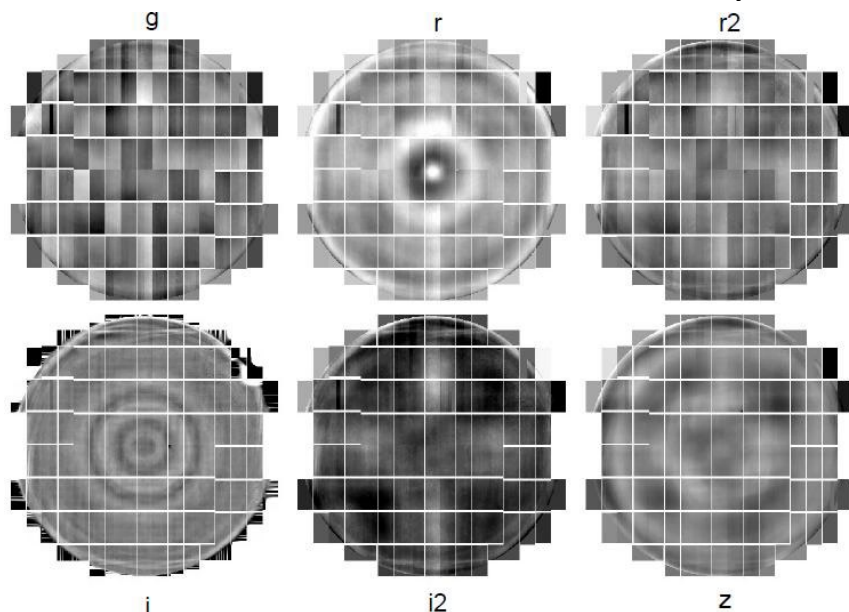
以前のHSC sky引きの問題: 明るい天体周りで引きすぎ



グローバルスカイ引き

大局パターン引き去り後の残差

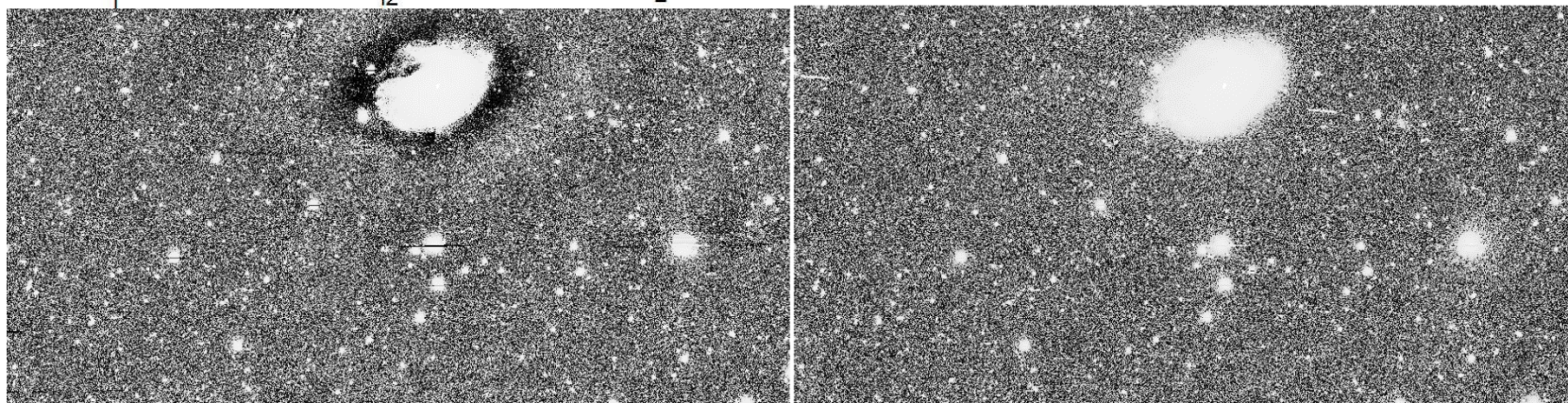
By Paul Price



- 各ショットごとにセルフスカイで128pxメッシュで背景を引くと問題となるような明るい・広がった天体のすそ野を保存することが出来る
- 十分なショット数を用いてスカイを作らないとノイズ源になる
- hscPipe8ではcoaddの天体検出直前に64pxメッシュでスカイが引き直される

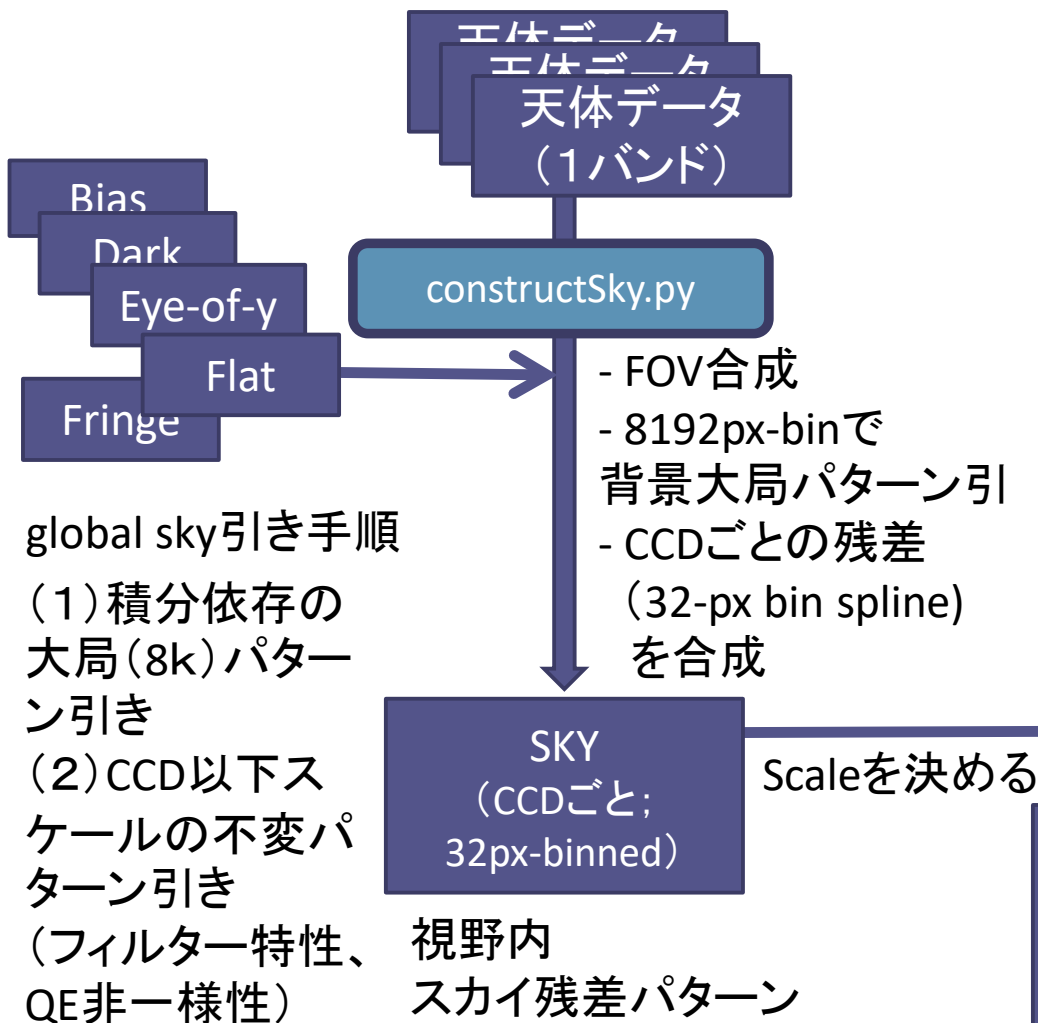
By Paul Price

グローバルスカイ引きの効果

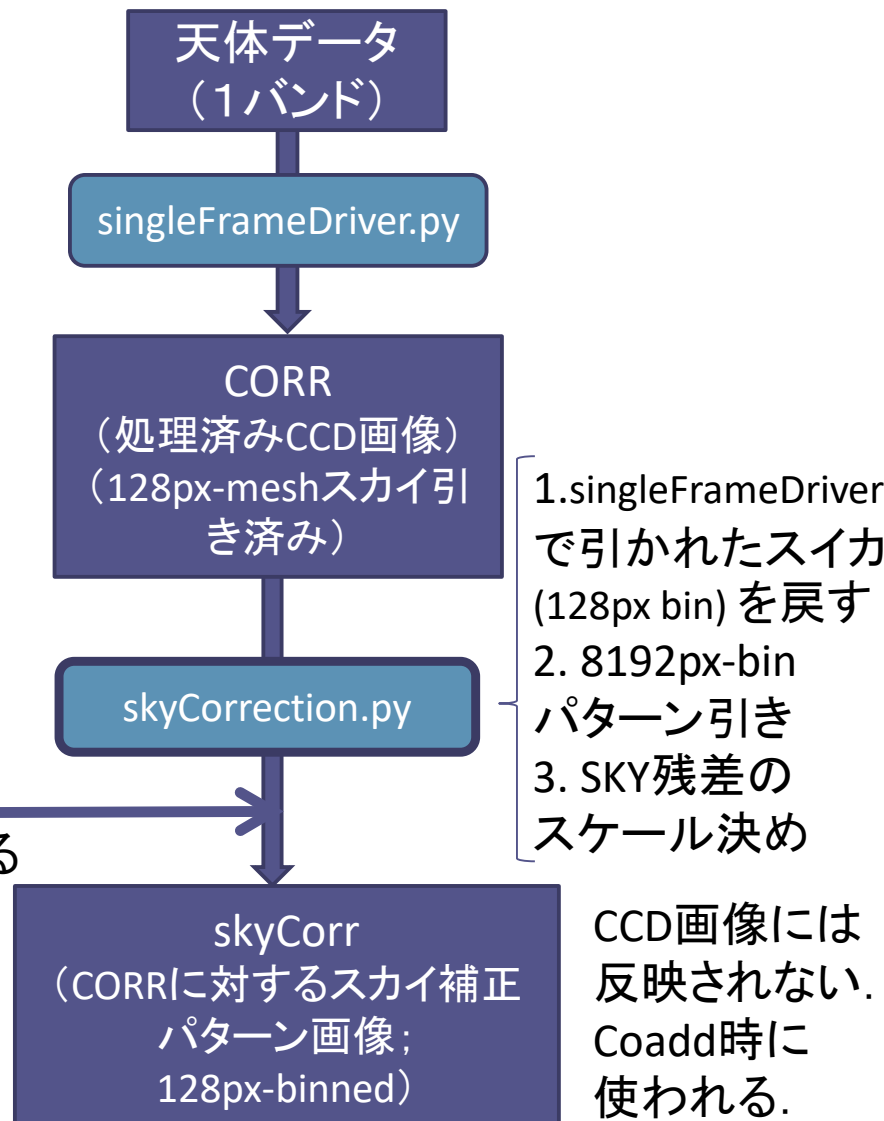


グローバルスカイ補正パターンの決定(バンドごとに行う)

スカイ残差パターン作成 (ほかのCALIBと同様の手順で作成)



スカイ残差引き



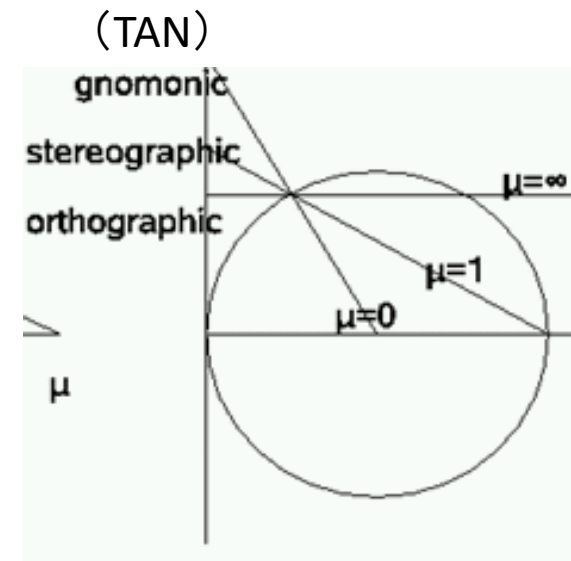
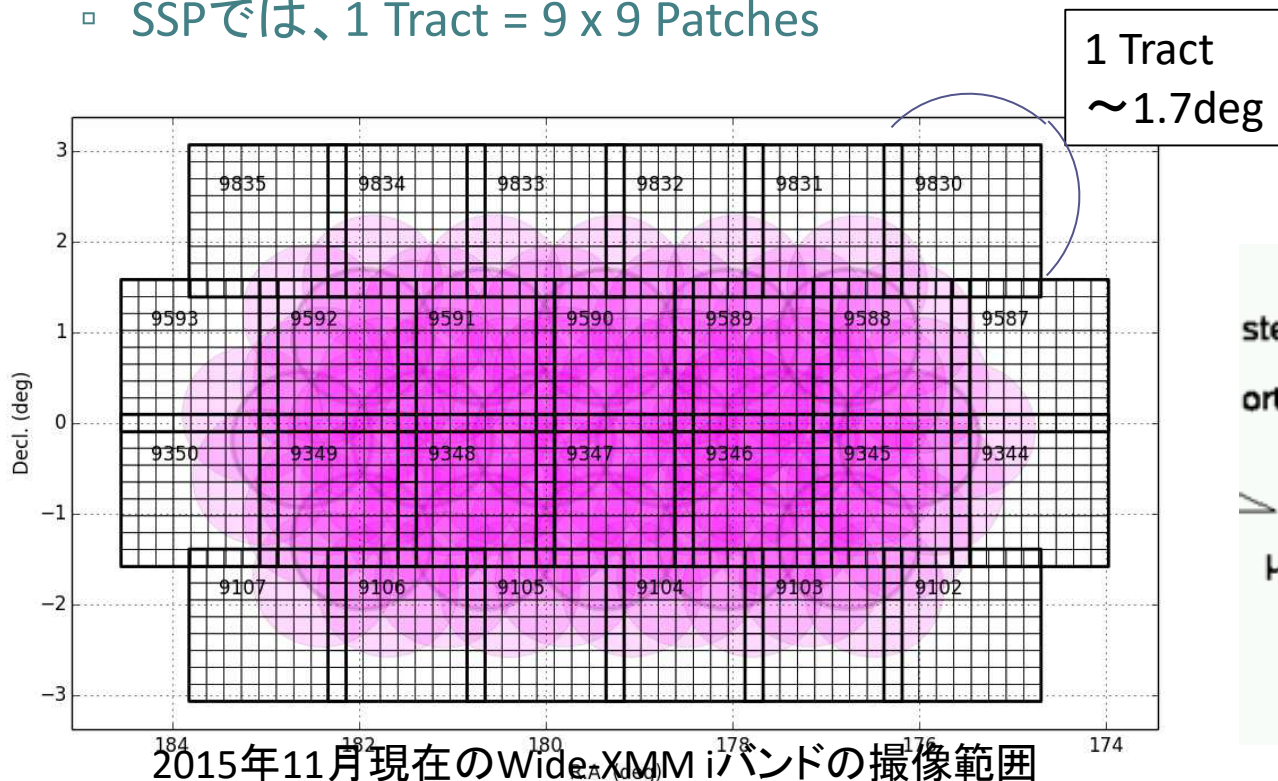
5. グローバルスカイ補正パターンの決定： constructSky.py → skyCorrection.py

- 利用データと手続き
 - (1)オブジェクトデータそのものを多数合成(セルフスカイ)し、各ショットの天体に影響されないスカイパターンを作成。
 - (2)各CCD処理済み画像に対してスカイ補正のための情報を作成
- **ステップ1: スカイパターン作成**
 - 手続き: オブジェクトCCDデータに対して通常のフラットフィールド・スカイ引きまで行い、天体マスク、8192pixビンによる大局パターン引きののち、残差パターン(32pixビン)をCCDごとに決め平均合成
 - 入力: CCD解析の結果画像(CORR) x理想は各CCD50枚以上。少なくとも>~20枚。
 - 出力: CCDごとのスカイ残差パターン=Image, mask, varianceを含むfloat32 FITS画像(SKY) (32ビン)
- **ステップ2: CORRに対するスカイ補正決定**
 - 手続き: 各CCD処理済み画像(CORR)に各ショットで決めたスカイを戻し、8192ビンによる大局パターン引きを行ったうえで、上のスカイ残差パターンにスケールし、スカイ補正を決定
 - 入力: CCDごとのスカイ残差パターン(SKY)、CCD解析の結果画像(CORR)
 - 出力: CORRから引くと正しいスカイになるようなスカイパターン補正画像 (128ビン) float32 FITS画像(skyCorr)
- **メモ**
 - SSPではランごとにスカイを作成。50ショット未満の場合は複数ランのデータを合成している
 - この操作では実際にはスカイ補正は画像に反映されず、Coadd時に反映される。
 - モザイクの結果は使われない

B. モザイクスタック解析・マルチ バンドカタログ作成

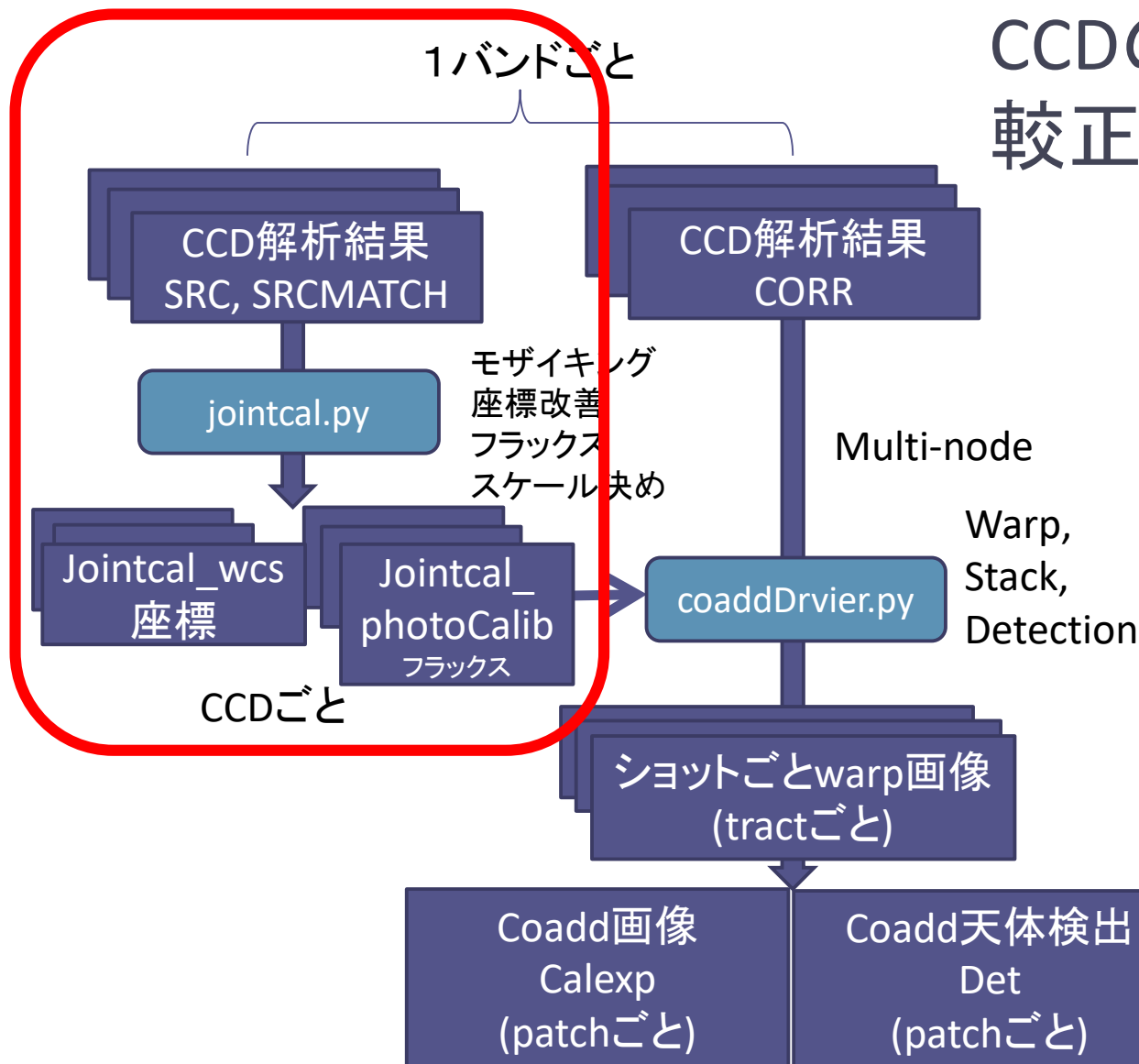
モザイク・ワープ・スタックとTract/patch

- Tract
 - モザイク・CoaddはTractに対して行われる(1.7度程度) → 事前に定義必要!
 - Tract内は共通の投影軸でTAN-WCSを共有
- Patch
 - Coadd画像・カタログを作る単位(4200x4200px, 隣接との重なり100pxずつ)
 - 解析ソフトウェアの都合で Tract の中を分割
 - SSPでは、1 Tract = 9 x 9 Patches



解析の流れ(2): モザイク・スタック

CCDの足しあわせと
較正の改善

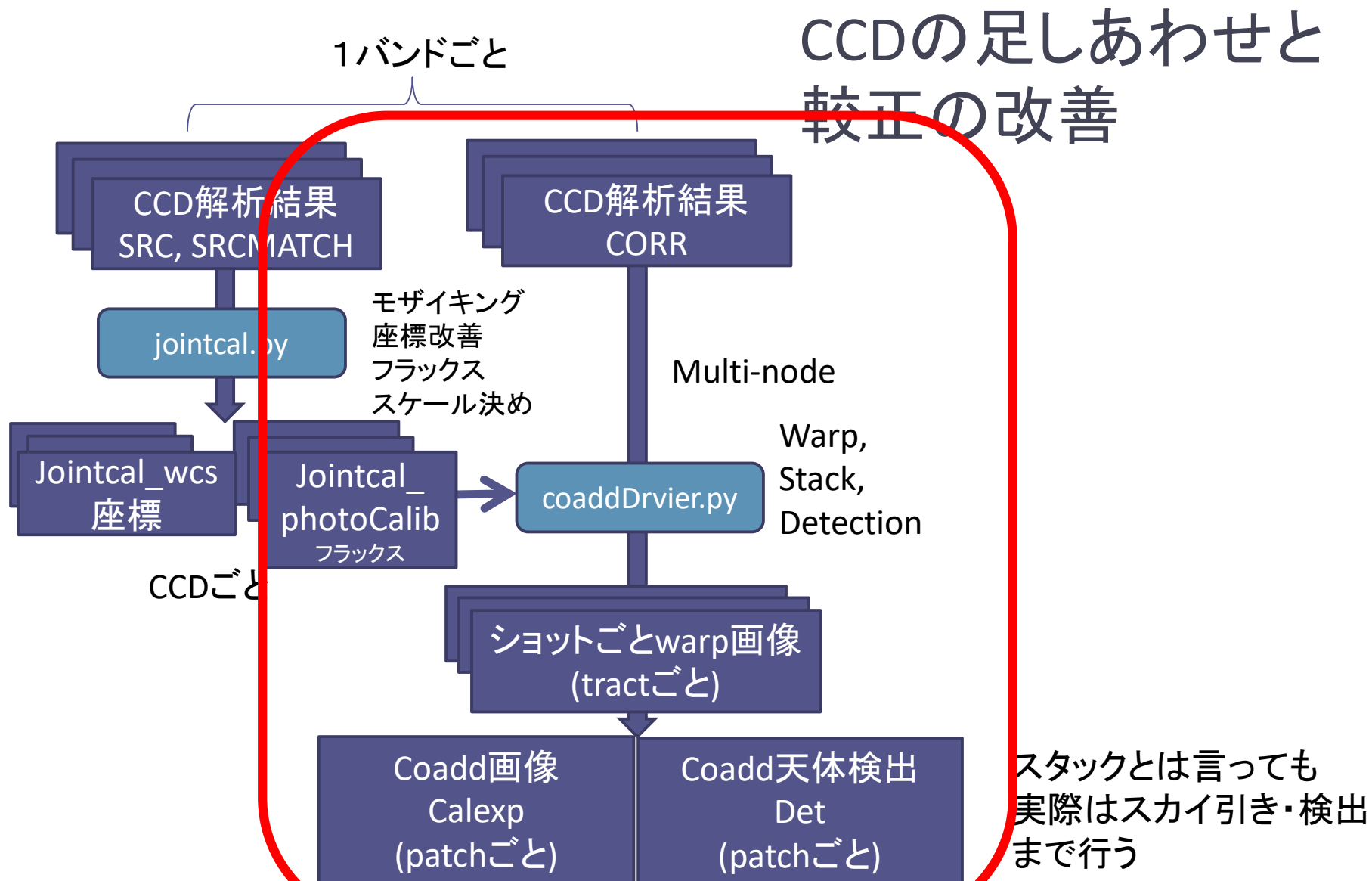


スタックとは言っても
実際はスカイ引き・検出
まで行う

1. モザイクキング (jointcal.py)

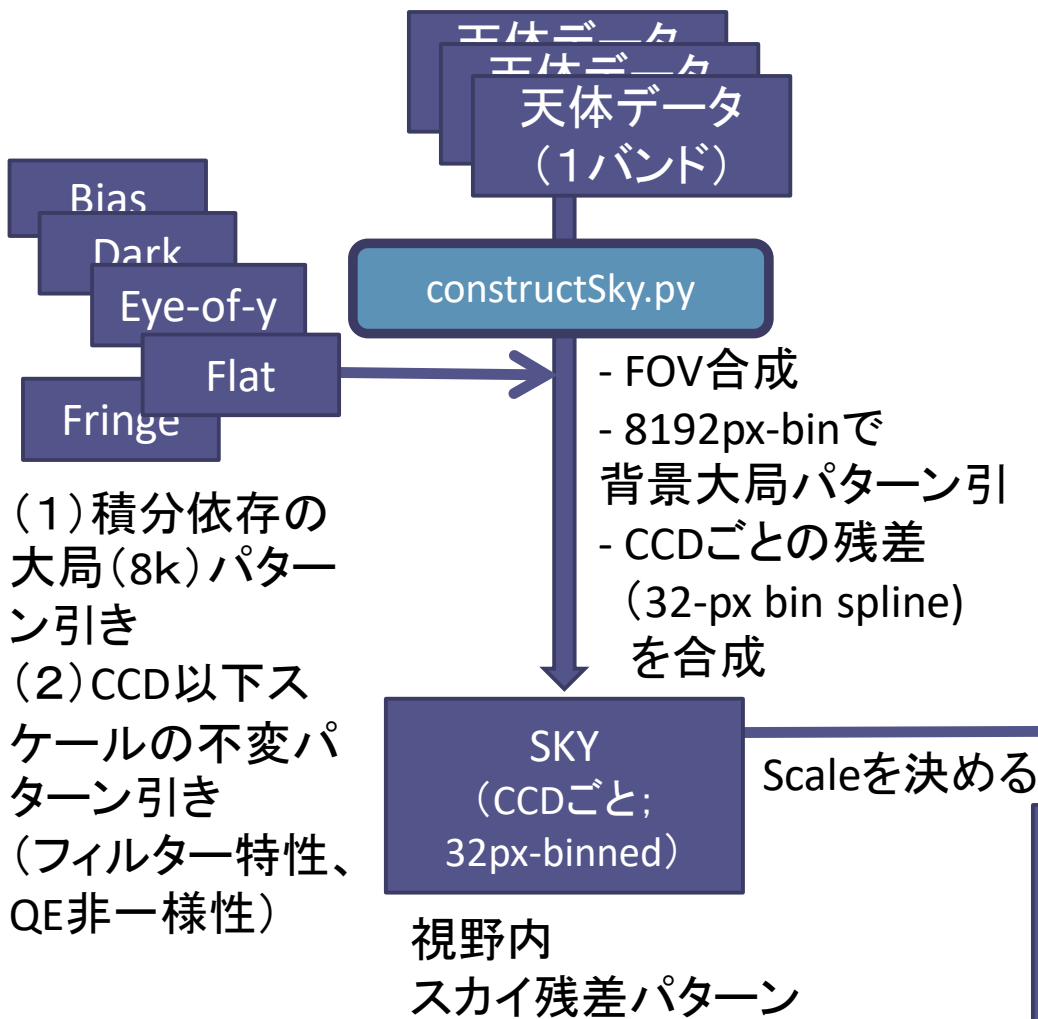
- 複数積分のCCDデータを合成するための情報出し
 - 各CCDの座標を合わせるための情報
 - ディザリングにより空の違う場所でデータが取られている
 - 複数の積分で写っている同じ星の座標、外部カタログの座標と合わせて使って、複数積分で測定した星の座標がなるべく一致するように、WCSを改善する
 - 外部カタログ座標を拘束条件に含めるため、天球座標は絶対値も合わせられる
 - 各CCDのフラックススケールを合わせるための情報
 - 積分時間や透過率が違う状況下でデータが取られている
 - 複数の積分で写っている同じ星のカウント、外部カタログの等級を合わせて使って、複数積分で測定した星のカウントがなるべく一致するように、CCDのフラックススケール+CCD面内の微小誤差を求める
 - 外部カタログ等級を拘束条件に用いるため、等級は絶対値も合わせられる

解析の流れ(2): モザイク・スタック

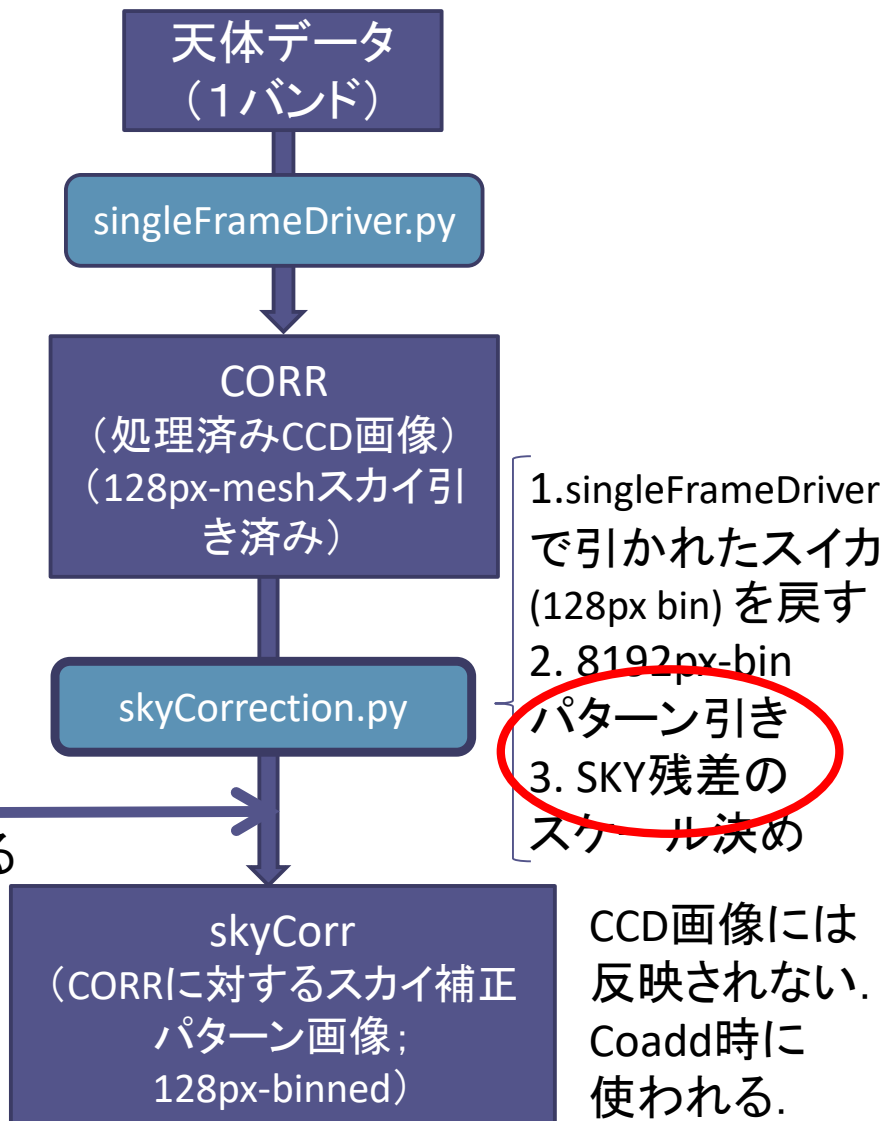


グローバルスカイ補正パターンの決定(バンドごとに行う)

スカイ残差パターン作成 (ほかのCALIBと同様の手順で作成)



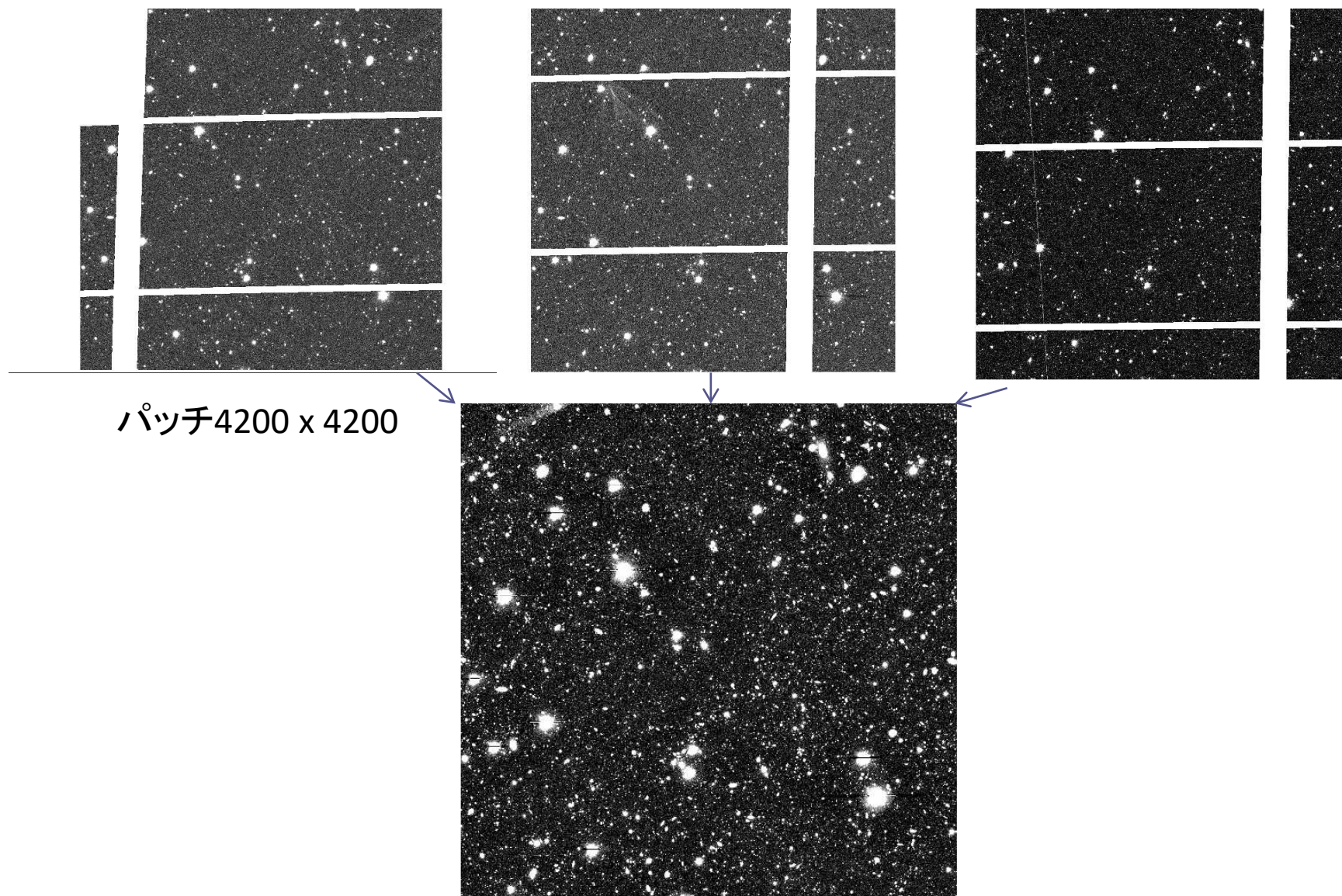
スカイ残差引き



2. ワープ・スタック: coaddDriver.py

- ワープ(パッチごとに行う)
 - モザイクングで決定したwcsとFcrを用いて、各CCDを画像変換する
 - 変換先は各パッチ
 - 出力1: warp (FITS画像+ α)
- スタック(=coadd)(パッチごとに行う)
 - 外れ値のクリップ
 - 各ショットのPSFを一致させたWarpとそれらのスタックの差分を取ることで疑似天体を除去(CompareWarpAssembleCoaddTask)
 - BrightObjectマスクのセットなど
 - 各ピクセルごとにmeanでスタック
 - 各CCDのPSFモデルも合成される(coaddPSF)
 - 出力2: calexp -最終天体検出に使われた局所スカイ引き後画像
\$(patch) -グローバルスカイ補正直後画像
※共にFITS画像+BINTABLE
- 天体検出(パッチごと)(DynamicDetection)
 - スカイ引き(128pix meshで天体マスクなどをして最終的に64pix meshでfootprintを決める)
 - 天体検出(天体の位置・領域だけを示したリスト作成)
 - 出力3: det (BINTABLE)、det_bkgd (FITS画像)

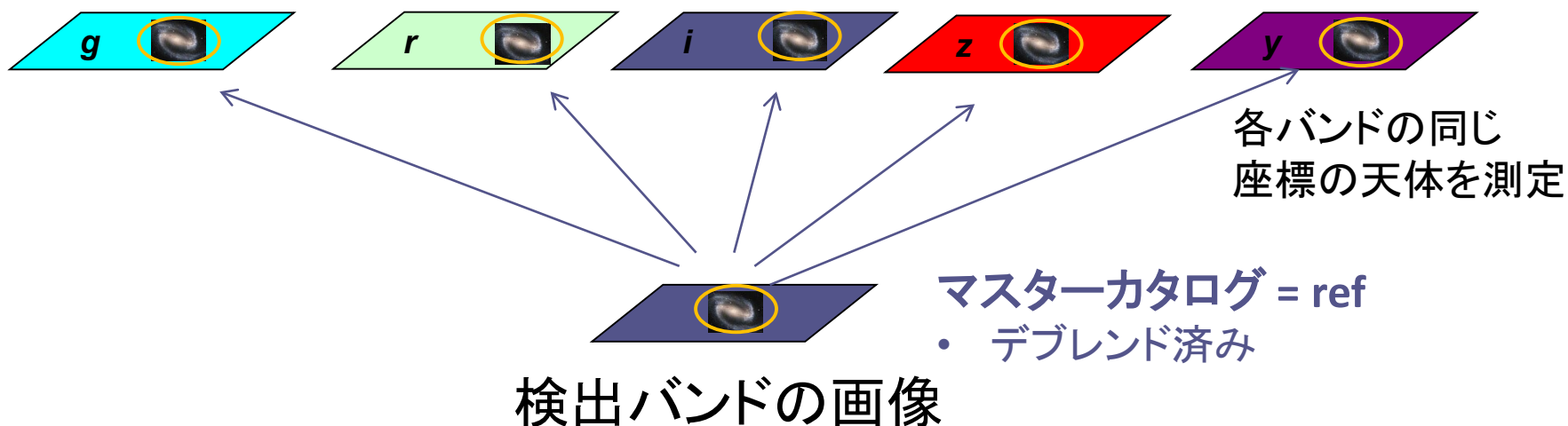
Warp (各ショット) とcalexp (スタック)



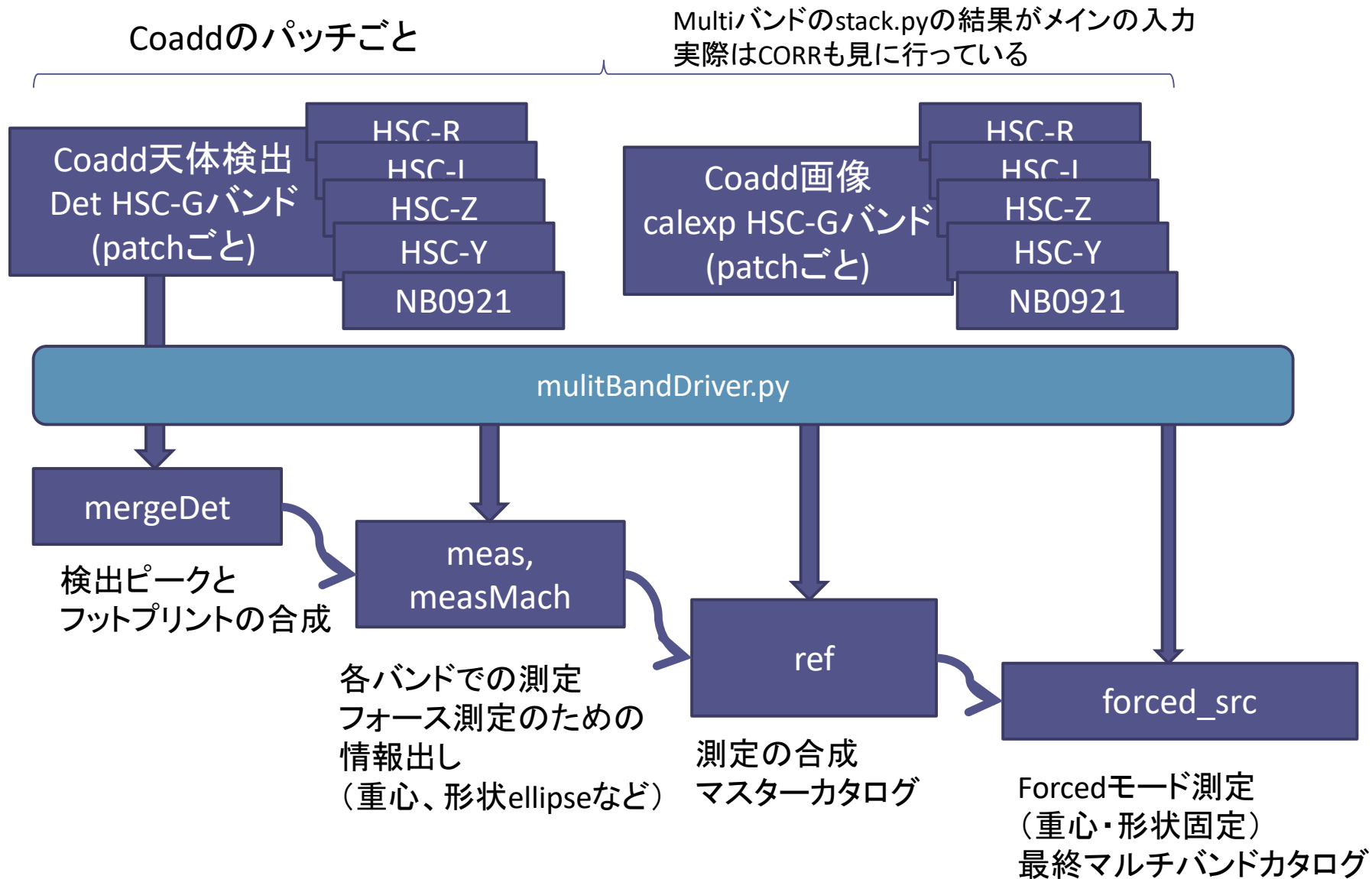
3. マルチバンドカタログ生成: multiBandDriver.py

- 手順

1. いずれかのバンドのCoadd画像で検出される全天体を含む**マスターカタログ**を生成
2. Forcedモード測定: マスターカタログの天体それぞれについて各バンドの画像を測定する

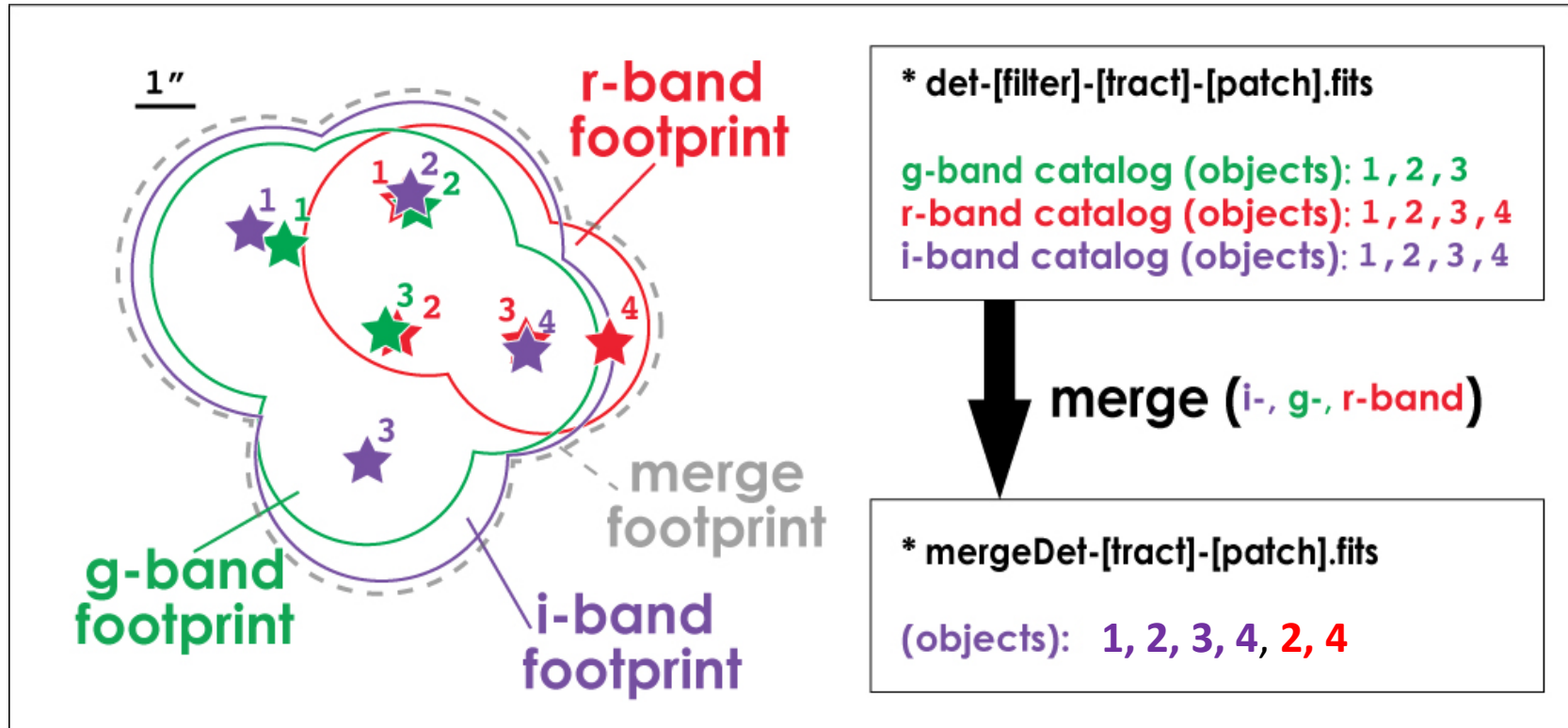


解析の流れ(3): マルチバンドカタログ生成



手続き1. 天体検出マージ

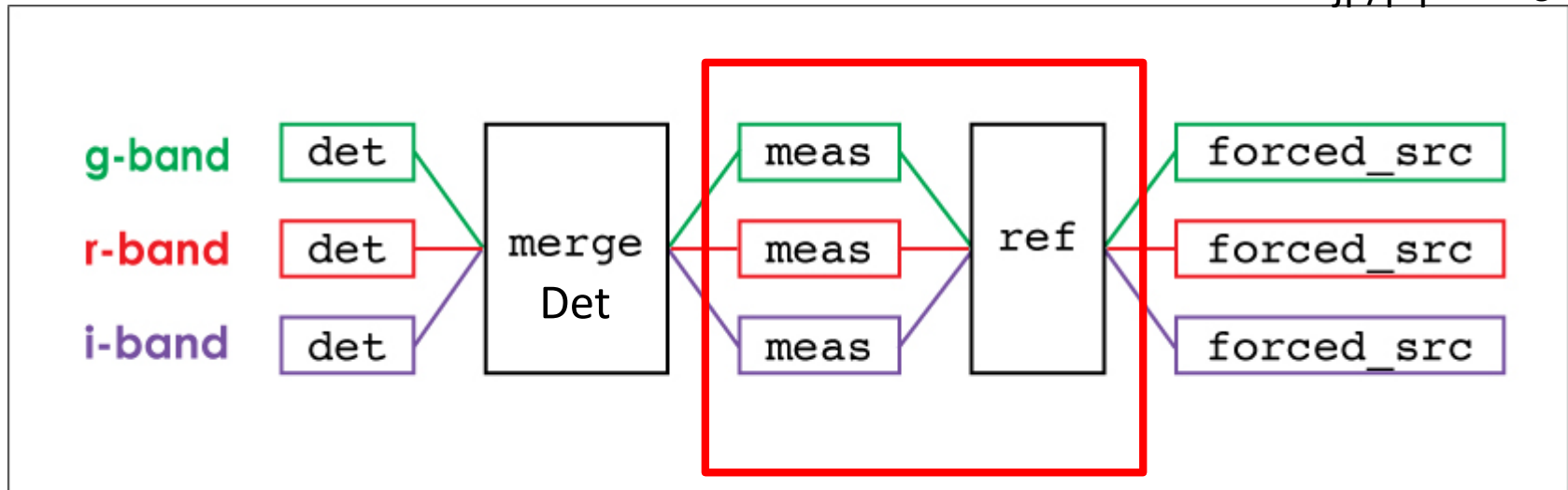
hsc.mtk.nao.ac.jp/pipedocより



- 各バンドで検出された天体のリストと各天体（デブレンド前）の占める領域（footprint）を拡大していく
- 複数バンドで検出されたピークを同一天体であると見なすルール
 - バンド間の距離が0.3秒以内の天体は同一とみなし優先順上位バンドの座標を採用、1秒以上離れていたら別天体として新規にリストアップする → 出力1: mergeDet
- 採択順位は、 $i \rightarrow r \rightarrow z \rightarrow y \rightarrow g \rightarrow n921 \rightarrow n816 \rightarrow n1010 \rightarrow n0387 \rightarrow n0515$

手続き2. マスターカタログ生成

hsc.mtk.nao.ac.jp/pipedocより



- mergeDetの各天体/footprintに対して各バンドのスタック画像を測定し直し、**デブレンド(各子天体のflux分割)**、重心座標、CModelのためのフィット範囲(exp, devのellipse)を決定
 - 出力2: meas
- measの上記値を再度、バンドの優先順位にしたがってマージ
 - 出力3: ref → これがforced測定のためのマスターカタログ
- forced測定 (refの各天体について、重心・形状固定で各バンドのスタック画像を最終測定)
 - 出力4: forced_src 各バンドごとに出来る

4-3. 天体検出測定部

- 検出

- スムージング

- PSFでスムージングして天体のピークを増幅する

- ピーク検出

- 局所的なノイズより十分大きいカウントを持つ
 - N-pixel以上繋がったところを探す

- 重なり合ったピークを持つ天体のフラックスの分離計算(デブレンド)

- 測定

- 位置(ピーク・重心座標 x, y)

- 形状(2次モーメント、楕円の方位角、銀河モデル化)

- 明るさ

- 開口(アパーチャ)測光

- 固定開口 : circular aperture flux

- 可変開口 : Kron flux

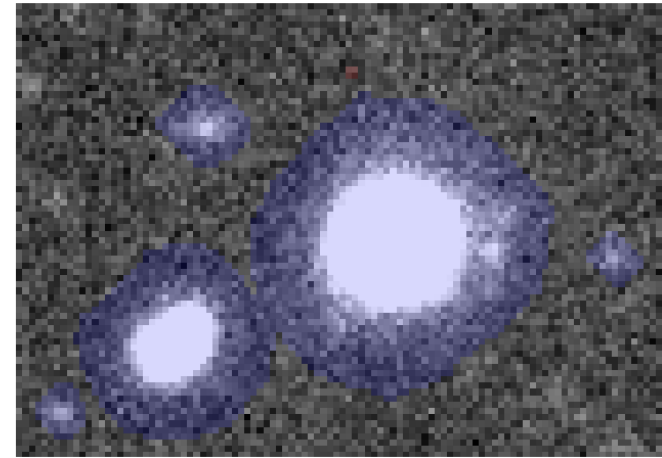
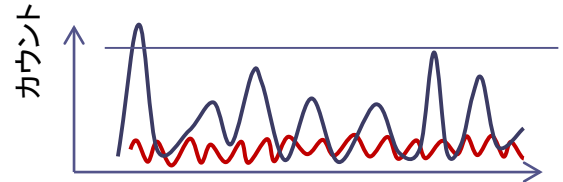
- モデル測光

- PSF flux

- CMoel flux – Composite Model flux = 銀河を2成分モデルでフィット

- フラッグ

- 天体測定したピクセルのマスク情報や測定中のイベントにより割当

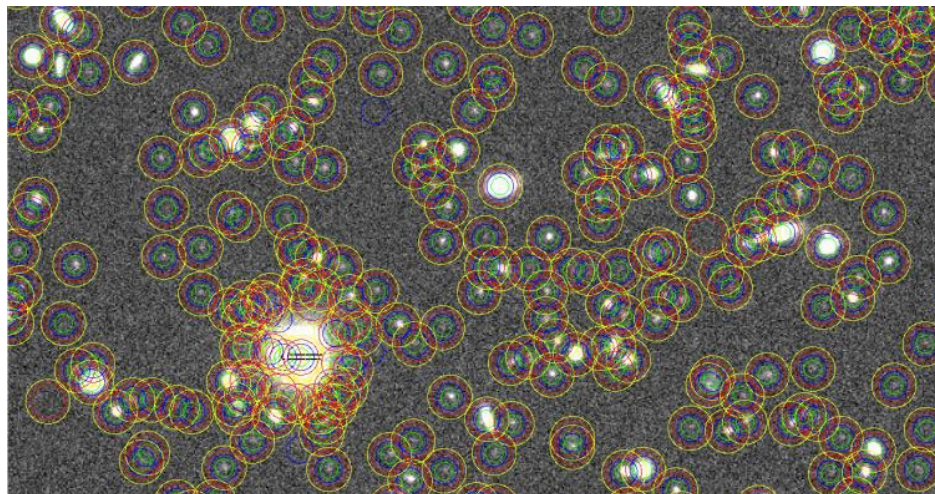


-0.044 -0.02 0.005 0.029 0.054 0.078 0.1 0.13 0.15

footprint

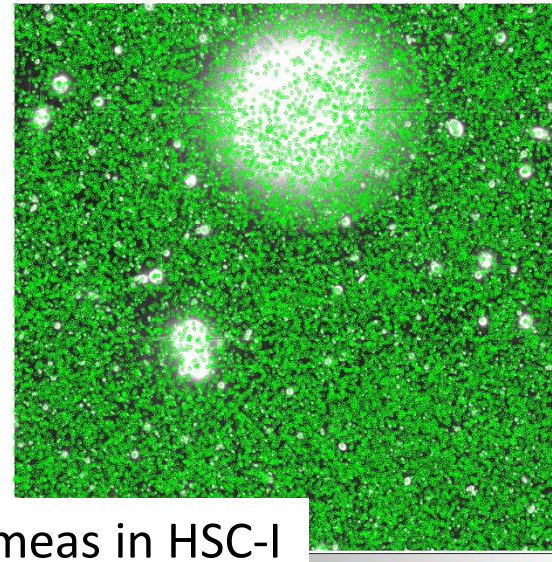
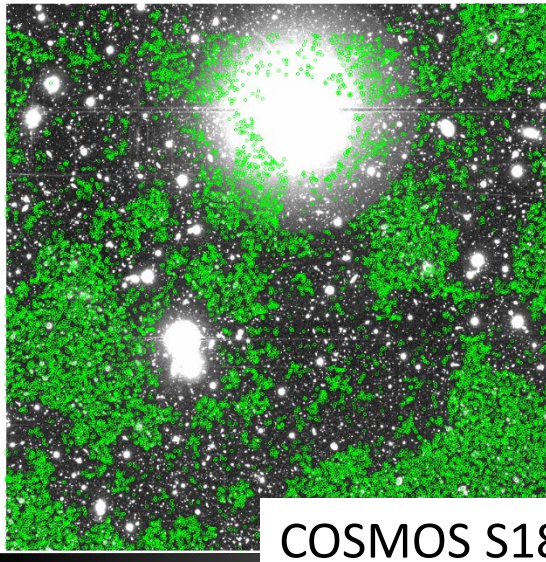
天体測定にはとても時間がかかる1. 混んだ領域のごみの除去 Peak Culling

- ごみらしき天体を事前に除去している
- ある天体ピークのブレンド集団 (footprint) 内の個々の天体をゴミと判定し落とすルール
 - 1バンドでしか受かっておらず、
 - 明るさ順で「30番目より」または「全体の7割かつ20番目より」暗い



天体測定にはとても時間がかかる2. 大きすぎるfootprintのデブレンドをスキップ

- 大きいfootprintの測定は大量のメモリと時間を要する
- デフォルトでは100万ピクセルを超える面積のfootprintの子天体は測定されない
 - `deblend.maxFootprintArea = 1,000,000`
 - すべてをデブレンドするためには以下を付けて実行
`--config deblendCoaddSources.singleBandDeblend.maxFootprintArea=-1`



COSMOS S18A meas in HSC-I

以下補足

新しいモザイクソフトウェア jointcal+FGCM

- hscPipe 7.9.1以降jointcalと呼ばれるLSSTチームが開発したより効率的なモザイクソフトウェア使われる
- しかし、hscPipe7~8時点では一部のデータ(ショット数が200を超えるデータや一部のNB)で測光精度が良くない(>~5% error)ことが分かり、これらの場合には位置較正(astrometry)だけに使う。
- さらに別のflux較正の手法FGCMが使える段階になったため、8.4以降の大規模なflux較正にはこれを使う。
 - Dark Energy Surveyを通して開発された手法->LSSTに向けて改修中->そのロードマップとしてHSCに適用。
 - <https://github.com/lsst/fgcmcal>
 - <https://arxiv.org/abs/1706.01542>

等級原点較正

- 込み入ったいくつかの注意点

1. HSCバンドシステムで等級較正

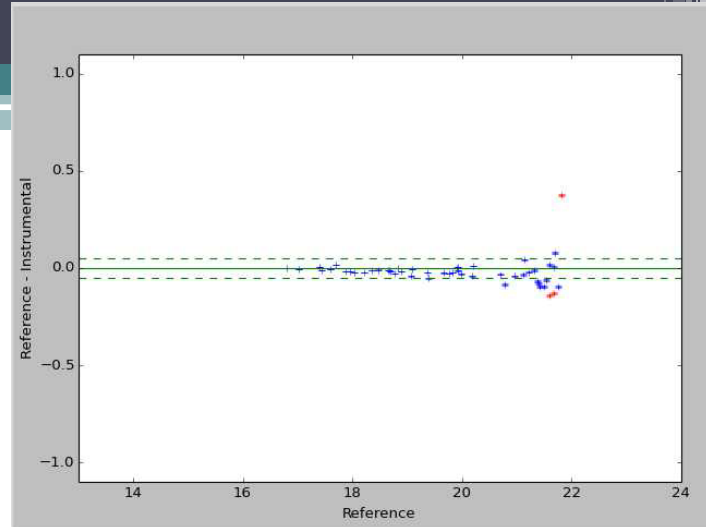
- 全CCDに対して同じと仮定されている
- CCD差、color terms、大気によるバンド変化は無考慮
- Airmassによる減光分は視野内の較正星を使うことで対応

2. 開口 $r=12\text{px}$ (直径約 4arcsec)

- 点源の全fluxを得るにはやや小さい
 - 等級原点はtotal magに対して若干明るく(値として小さく)なる
- Total magを得るにはアパーチャ補正が必要
 - hscPipeでは、Aperture flux 以外の測定では $r=12\text{px}$ flux への調整分に対応するアパーチャ補正がかかる

3. ディストーションによる輝度変化の影響は無補正

- 厳密には、この時点では、視野端にいくほど等級原点は明るくなる(最大~7%)が、CCDごとには一つの値を採用している
- CCD画像をSExtractorなどで測定する場合は、2のアパーチャ補正の効果と合わせて注意



カタログにリストされる天体パラメータ

スキーマブラウザとデータリリース論文, パイプライン論文等で確認しましょう

- スキーマブラウザ

<https://hsc-release.mtk.nao.ac.jp/schema/>

https://hscdata.mtk.nao.ac.jp/schema_browser3/

- 論文

Aihara et al. 2017 PASJ, 70, S8 SSP public data release (PDR1)

Aihara et al. 2019 PASJ, 71, 114 SSP public data release (PDR2)

Aihara et al. 2022 PASJ, 74, 247 SSP public data release (PDR3)

Bosch et al. 2018 PASJ, 70, SP1.S5 (パイプライン論文)

- ID・座標
- 形状
- Flux・等級
- Flags

天体パラメータ： ID、座標

- id -- 天体・各バンド・Tract・PatchごとにuniqueなID
- coord_ra, coord_dec
- 重心
 - base_SdssCentroid_{x, y}{Err}
 - ピーク周り 3x3 pixの中の多項式フィット
 - base_SdssShape_{x, y}{Err}
 - 楕円領域内の重心

天体パラメータ：形状・サイズ関係

- モーメント
 - `base_SdssShape_{xx,yy,xy}{Err}` – SDSS adaptive moment
 - 2nd momentとして利用
- Kron半径
 - `ext_photometryKron_KronFlux_radius` – $\sqrt{a*b}$
- Galaxy Component-Model Fit (CModel)
 - `modelfit_CModel_fracDev`
 - $\frac{1}{4}$ 乗則とexp lawの2要素のプロファイルフィット結果の比率

天体パラメータ： フラックス・等級測定

- 固定円アパーチャフラックス
 - 各バンド測定カタログ(単位はDBテーブルによる)
 - `base_CircularApertureFlux_{X_X}` (pix表記)
 - $\phi 1, 1.5, 2, 3, 4, 5.7, 8.4, \dots, 23.5''$
- PSFフラックス(以下はマルチバンドテーブルを割愛)
 - `base_PsfFlux_instFlux{Err}`
- CModelフラックス
 - `modelfit_CModel_instFlux, modelfit_CModel_{dev, exp}_instFlux`
- Kronフラックス
 - `ext_photometryKron_KronFlux_instFlux{Err}`
- Footprintを分割しない(undeblended)状態でのPFS合わせ固定円アパーチャフラックス
 - デブレンドが苦手な混んだ領域の天体のカラーを知りたいときに、オプションとして使うと良い
 - `undeblended_ext_convolved_ConvolvedFlux_0_6_0_instFlux` など

目的に応じたフラックスの使い分け

- PSF = 点源
- CModel = 広がった天体・広がりのよくわからない天体
- カラー
 - 天源 -> PSF
 - 広がった天体 -> CModel

目的に応じたカタログの使い分け

- forced_src
 - マルチバンド測定値として基本はこれを使う
 - 測定位置 (CModelについては初期値) は全バンド固定
 - 必ずしも厳密に欲しいバンドの天体ピーク位置に一致しない場合はある
- meas
 - 各バンドでの天体ピーク位置での測定、また天体のピークの座標が重要な場合はこれを使う
 - 各バンドごとに、そのバンドで写っている天体ピーク位置で測定されている

フラッグ: 検出状態1

- `detect_isPrimary (ref)`
 - 同じ天体に対して複数の測定がある場合に、主とみなすべき測定
 - 特に, `coadd`カタログには`skyObject`(擬似天体)が含まれるので, `detect_isPrimary=True`により除外する必要がある
- `detect_isTract_inner (ref)`
- `detect_isPatch_inner (ref)`
 - 隣接する`Tract`・`Patch`画像の重なり領域にある天体について、より`Tract`・`Patch`の中心に近い方に位置しているか
- `デブレンド (meas, forced_source, ref)`
 - `parent`
 - 自分が分割された後の子の場合、分割前の天体IDが入る。そうでない場合はゼロが入る
 - `deblend_nChild`
 - 自分自身が複合天体である場合、分割された子供がいくつかあるか。自分自身がこれ以上分割できない天体の場合はゼロが入る。普通はこれがゼロである天体に注目すればよい

Patch-Inner, Tract-Inner, Primary

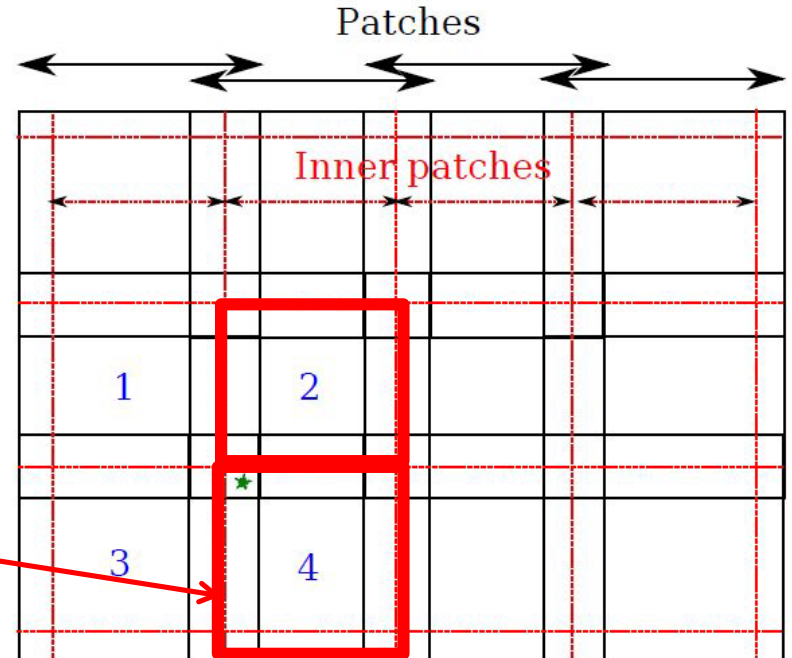
- Tract間、Patch間の重なり → 同一天体の重複測定
 - Tract間～1分角、Patch間～200px(~34")
- Primary天体
 - 主要な測定を示す → マルチTract・Patchから重複なく天体リストを作れる
- より中央に近い位置に重心がある天体がInnerでありPrimary天体

Patch-Inner:

4000 x 4000 px の内部領域

Patch-Outer:

その外側



フラッグ: 検出状態2

- base_PixelFlags_flag_bad (*)
 - 事前定義されたdefect領域にある
- base_PixelFlags_edge
 - Coaddの視野外にある・Coaddできるショットが存在しない
- base_PixelFlags_offimage (*)
 - 画像(FITSピクセル)の外部にある
- base_PixelFlags_cr{Center}
 - Cosmic Rayが検知された
- base_PixelFlags_interpolated{Center} (**)
 - カウントがピクセル間でInterpolateされた
- base_PixelFlags_flag_saturated{Center} (**)
 - 生データでOverflowしている(そのようなショットが卓越している)
- base_PixelFlags_suspect{Center} (*)
 - カウント線形性が怪しまれる(生データで50000ADUを超えている)
- base_PixelFlags_bright.object{Center} (**)
 - 明るい星(NOMADカタログ)の近傍にある
- base_PixelFlags_inexact_psf{Center}
 - Coadd時にPSFが正しく決められなかった

(**) Centerは中心付近のみ影響を受けている。

(*) coaddで伝搬しない

通常はcenterのみの指定で十分

フラッグ：測定状態（Trueで異常あり）

- 重心
 - `base_SdssCentroid_flag`
- 形状
 - `base_SdssShape_flag`
- フラックス
 - `ext_photometryKron_KronFlux_flag`
 - `base_PsfFlux_flag`
 - `modelfit_CModel_flag`
- デブレンド（あるfootprintがデブレンドされたかどうか）
 - `deblend_skipped`, `deblend_tooManyPeaks`,
`deblend_parentTooBig`

フラッグ：天体の特性・付加情報

- PSFと比べて有意に広がったプロファイルを持つか
 - `base_ClassificationExtendedness_value`
 - PSFと比べてCModel(*fac)が明るければ広がっている=1
- i/i_2 , r/r_2 の割合
 - `subaru_FilterFraction_{weighted/unweighted}`

フラッグ：multiBand処理の状態（上級）

- そのバンドでピーク検出されたかどうか
 - `merge_peak_{g,r,i,z,y,...}`
- そのバンドのfootprintが合成されたかどうか
 - `merge_footprint_{g,r,i,z,y,...}`
- そのバンドの測定がマスターカタログ（forced測定の初期値）に採用されたかどうか
 - `merge_measurement_{g,r,i,z,y,...}`
- マスターカタログ作成時の優先順位
 - `i → r → z → y → g → n921 → n816 → n1010 → n387 → n515`

3. フラットフィールドディング (3)

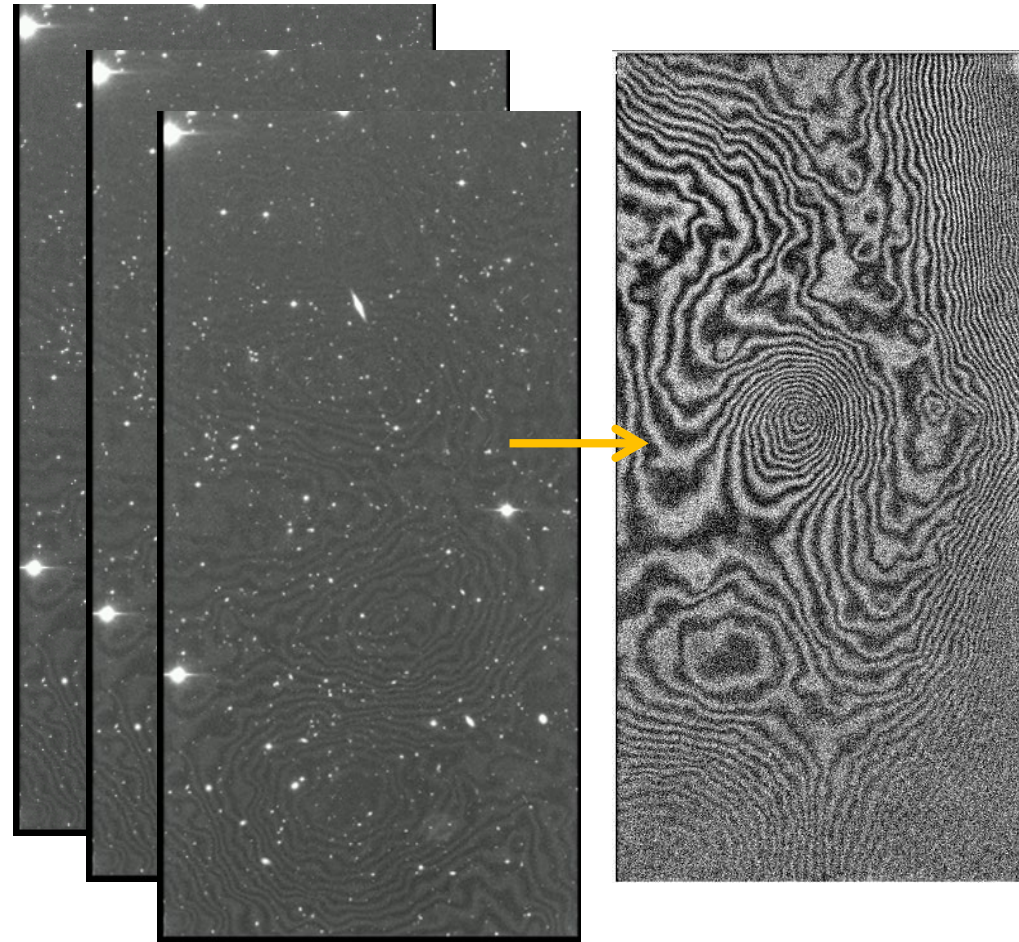
- 望遠鏡から検出器 (CCD) までの正味の感度ムラにより、同じレベルの入射信号に対して装置視野の場所ごとに受信カウントが増減した分を2Dで割って戻す
- 乱暴には、カメラ視野 (or CCD) に対してほぼ固定の感度パターンだと思ってよい
 - アンプゲイン・光学けられ、次にフィルター・CCD QEが主要因のため
- 感度の絶対値までは決まらない
- 考慮すべきことは色々ある (以下メジャーなところ)
 - Distortionによる輝度変化の効果は正しく直せない
 - ・ フラット後の輝度は平らにはなるけどそれは正しくない
 - 視野内・ピクセル間にバンド差 (カラーターム) がある場合、その効果は直せない
 - ・ フラット光源とバンド内で同じスペクトルを持つ天体だけ正しく補正される
 - フラットパターンは厳密には視野・CCDに固定されていないことが多い

3. フリンジ作成 (y, NB {0921, 0926, 0973, 1010})

- 利用データ
 - フリンジを補正したい夜ごとのオブジェクト・night sky データ
 - ・ 天体が混みすぎず、かつ大きな天体の含まれない20ショット以上程度あることが望ましい
 - ・ フリンジは主にOH夜光がCCD面で作る干渉縞と考えられている。ドームフラットやトワイライトは通常使わない
- 手続き
 - 各CCD: オーバースキャン引き・バイアス引き・ダーク引き
 - 全体: 各CCDの天体マスク、スカイレベルで規格化(← フリンジはスカイレベルに比例すると仮定)し、スカイ引き、メディアンスタック
- 入力
 - ナイトスカイフレーム x~20 (、バイアス、ダーク)
- 出力
 - 各CCDごとに1 HDUのfloat32 FITS画像
- メモ
 - y, NB{0921, 0926, 0973, 1010} 以外のバンドのフリンジ引きはソフトウェア設定を変更しないといけないので、helpdeskに相談して下さい

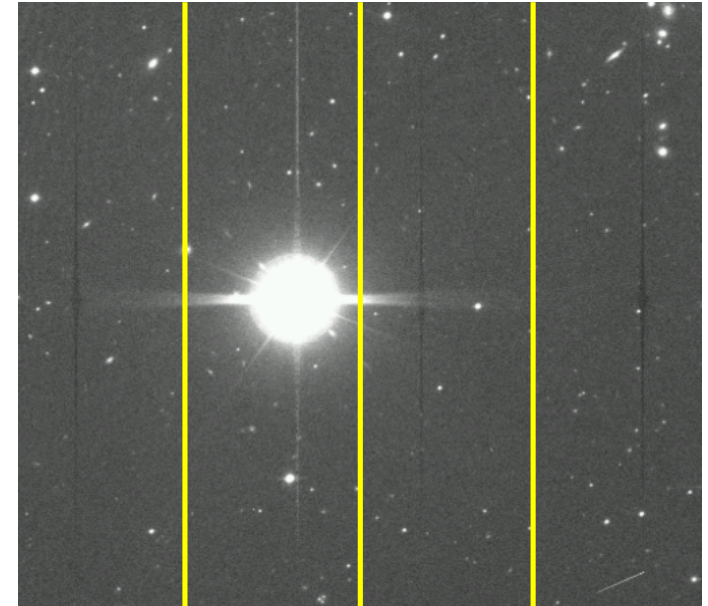
フリッジとフリッジ補正

- フリッジはCCD境界面での干渉により起こる縞模様。主にOH夜光の影響だと言われており、赤い波長帯で顕著
- 一般に
 - 時間・空の状態、装置の姿勢によって変わるかもしれない
 - QEが高ければ起きづらい
- SSPでは、
 - Y, NB0921のみ対応。天体画像のスタックによりYは夜ごと、NB0921はランごとのFringeを作り、パターンは不変、強度だけをオブジェクトの実測値とスケールして2Dで引いている
 - $f = 3\text{px (fringe)} - 30\text{px (sky)}$
 - 30000ランダム位置で測定



クロストーク

- 本来独立であってほしい読み出し(アナログ)回路のアンプ間で電位に相関が出ること
- 現象
 - 明るい天体があつたりすると、アンプに対して対称な位置に人工的なカウント増減が出る
 - 天体信号起因のクロストークは、アンプ対称性のため、PAを変えないディザリング+スタックだけでは消えず、クロストークのS/Nが上がってしまう

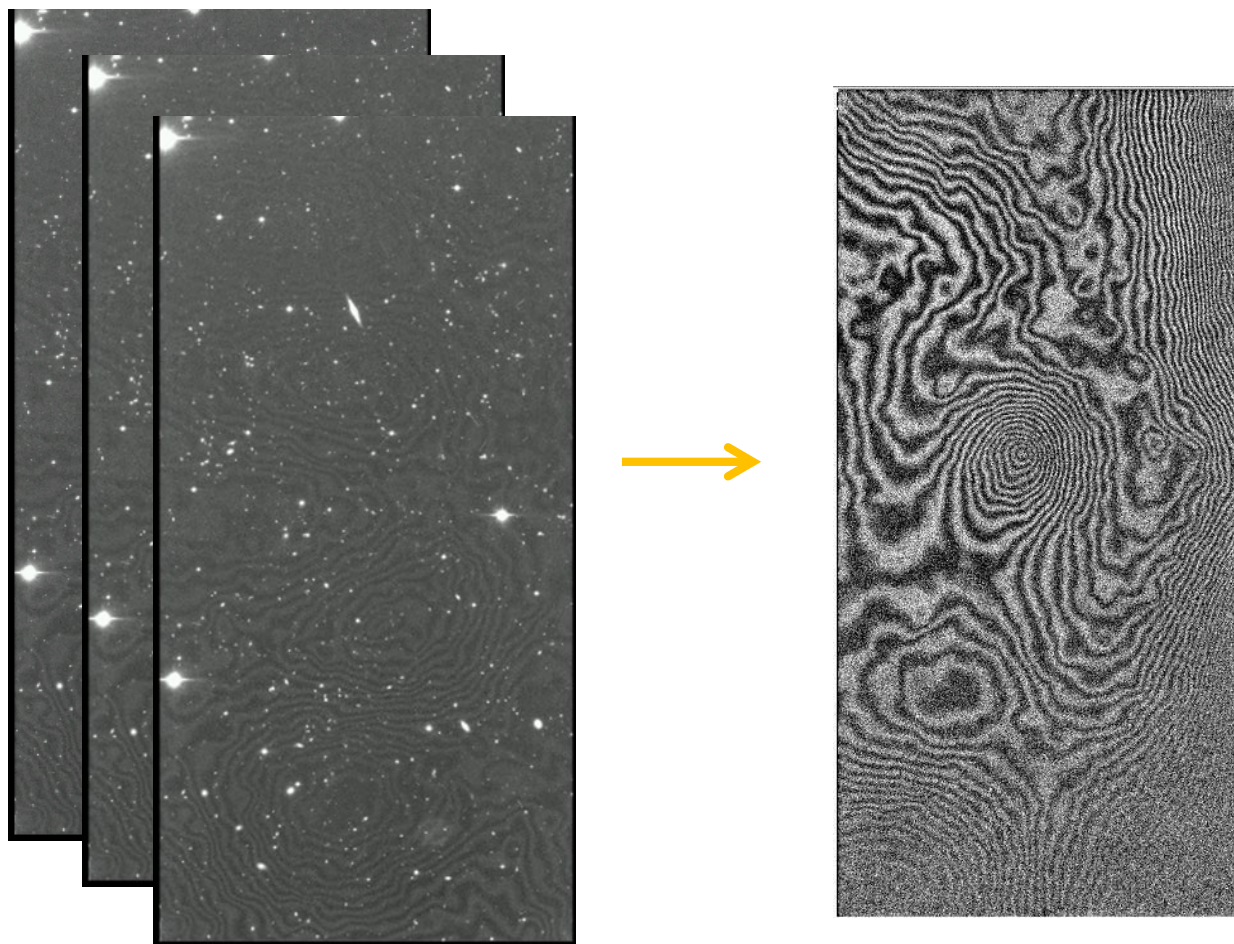


HSC CCDの典型的なアンプ間
cross talk

クロストーク補正(2)

- SSPでは
 - HSCやSCamのCCDでは、原因信号に対して各チャンネルのX軸方向に対称な位置(CCD内の3か所)にnegativeのcross-talkが出る
 - CCDのチャンネルごとに決められた係数を使い(原因信号 * $\sim 10^{-4} \sim -5$ 程度)補正している
 - 局所的な測光に影響が出る。特に26.5等級より暗い天体ではスカイ引きを通して数10%以上の測光誤差になり得る
 - 係数は `$OBS_SUBARU_DIR/config/hsc/isr.py` にある

フリンジ作成

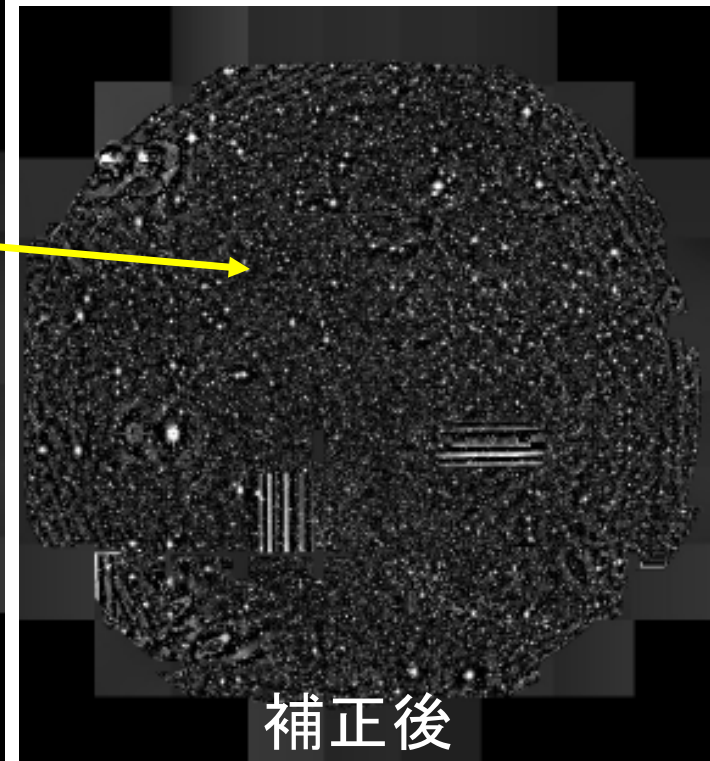
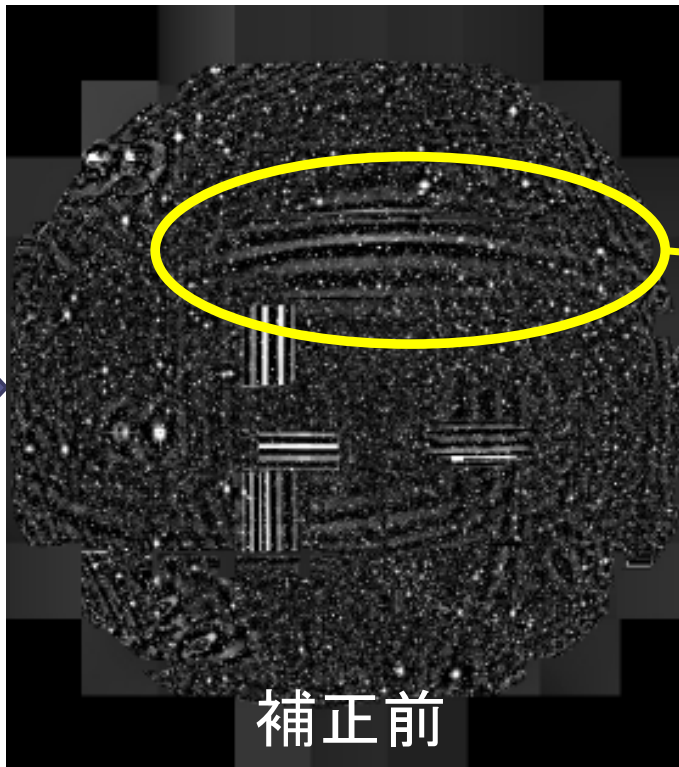
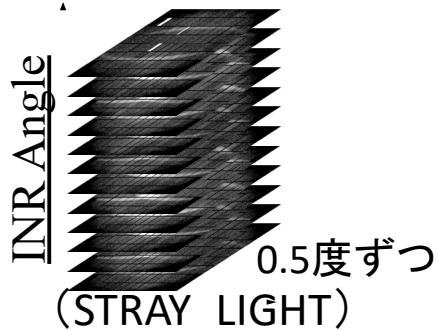
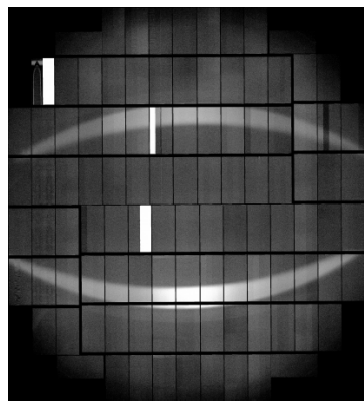


4-1. CCD解析： 画像処理パート

- サチュレーション・バッドピクセルへのマスク設定
- ★ オーバースキャン引き & 領域トリム
 - 30点スプラインで補間しラインごとに差し引き
- Variance生成
 - 画像カウントとゲインから。電荷のポアソン統計・ピクセル間独立を仮定
- ★ (バイアス引き)
- ★ 線形性補正
 - 各CCDのチャンネルごとに1次補正: $f = f_0 + (1.0e^{-7\sim-8}) \times f_0$
- クロストーク補正
 - CCD間に差はないと仮定。チャンネルごとに1次補正: $df = \sim 1.5e^{-4} \times f(\text{sig})$
- Brighter-flatter (高カウントほど電荷が周囲にしみ出す効果)補正
- ★ (ダーク引き)
- ★ 散乱光 (eye of y) 引き (yバンドのみ、NB0973は試験中)
- ★ フラットフィールドでの割り算 (フラットフィールドディング)
 - 欠損ピクセルBADマスク & 内挿
 - 事前定義のリストに従う
 - サチュレーション・オーバーフロー内挿
 - フリンジ引き

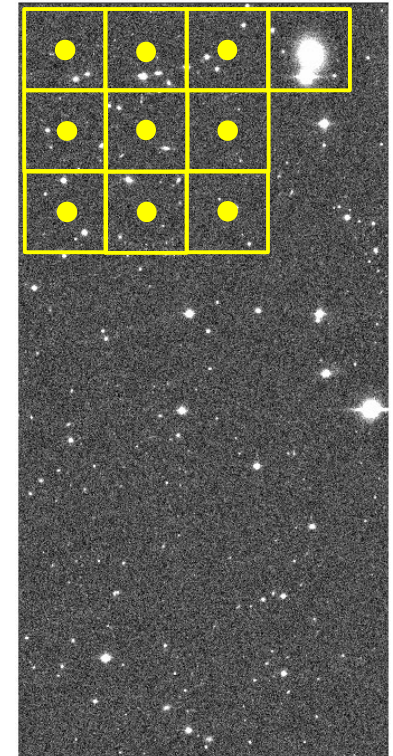
散乱光 (eye of γ) 引き

- 装置ローテータのエンコーダに起因する迷光
- 2018年1月にカメラ側で遮光された
- ccd=9全体, 33 (amp=0,1), 43 (amp=0) は取り扱わず



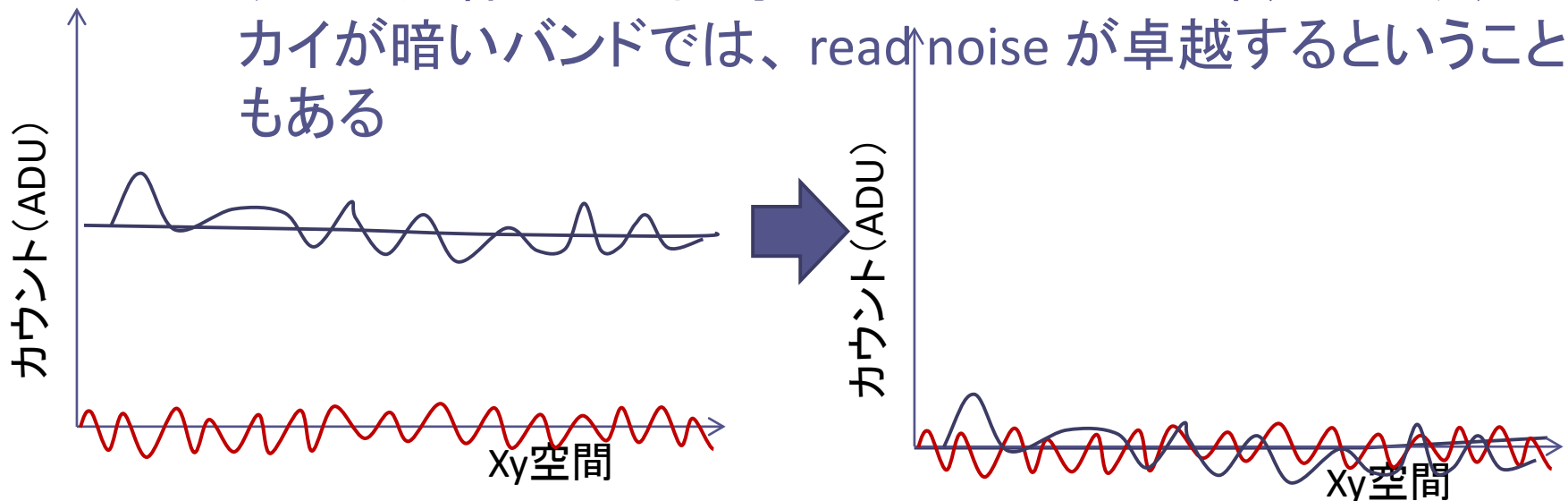
4-2. CCD解析 : Calibrateノパート

- スカイ引き1 (128pix mesh, 6次Chebyshev補間)
- 較正に使う天体検出
 - 10σ
- PSF測定
 - アstrometry-外部カタログと星マッチ
 - 等級原点決定
 - 上を入力としてPSFをCPA(主成分解析によりモデル化)
- 宇宙線 (CosmicRays) 除去
- スカイ引き2 (128pix mesh, 6次Chebyshev補間)
 - 天体マスク済・CCD解析中のメインのスカイ引き
- アパーチャ補正量の決定
 - 天体測定
 - PSFモデルとのフラックス比を各測定アルゴリズムごとに求め、CCD内の座標の関数としてアパーチャ補正量を決定
- 較正に使う天体測定
 - Aperture fluxes以外はアパーチャ補正を適用
- アstrometry
- 等級原点決定



5. スカイ(バックグラウンド)引き

- 夜空は明るいので、CCDに結像するカウントの大部分はスカイバックグラウンド(背景光)
 - ピクセルごとのスカイのポアソンノイズが、暗い天体に対しては測定時のノイズとして卓越する
 - 明るい天体はFlux自身のPoisson noiseが卓越したり、スカイが暗いバンドでは、read noise が卓越するということもある



PSFの測定

- Point Spread Function (点源が画像上でどういう形状になっているか)
- 光学系、カメラ・望遠鏡制御、シーイング(大気)の組み合わせで決まる
 - 一般にGaussianよりは裾が広い。軸対称とは限らない
- SSPでは、主成分解析PCAによるモデリングを場所ごとに行い、係数を多項式フィットしている
 - 現在はastromatic.netのPSFEx のエンジンを改良したものをデフォルトで使っている

FLUXMAGOの確認

- hscPipeのAPIを使ったオブジェクト操作で確認

```
>>> import lsst.afw.image as afwImage
>>> exp = afwImage.ExposureF(filepath_to_CORR)
>>> pc = exp.getPhotoCalib()
```
- FLUXMAGO

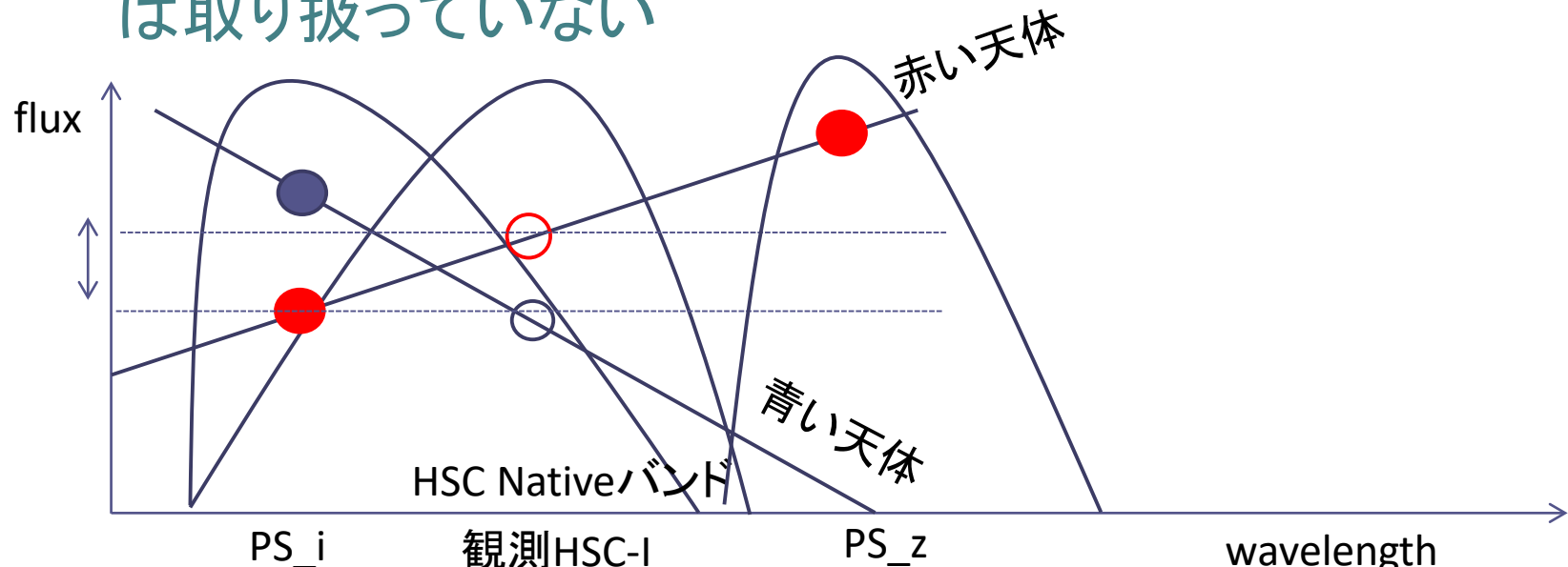
```
>>> FLUXMAGO = pc.getInstFluxAtZeroMagnitude()
```
- FLUMAGOERR

```
>>> ref_flux = 1.0e23*1.0e9*10**(-0.4*48.6)
>>> FLUXMAGOERR = FLUXMAGO *
pc.getCalibrationErr()/pc.getCalibrationMean()
```

※astropyなどでFITSデータ内を探るのは少々たいへん

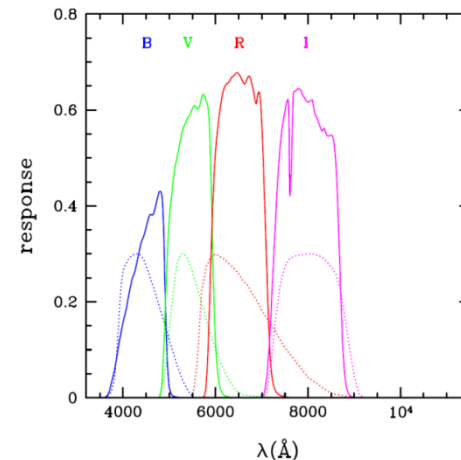
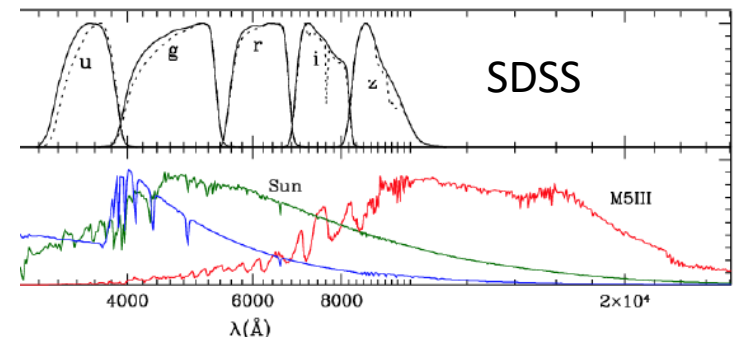
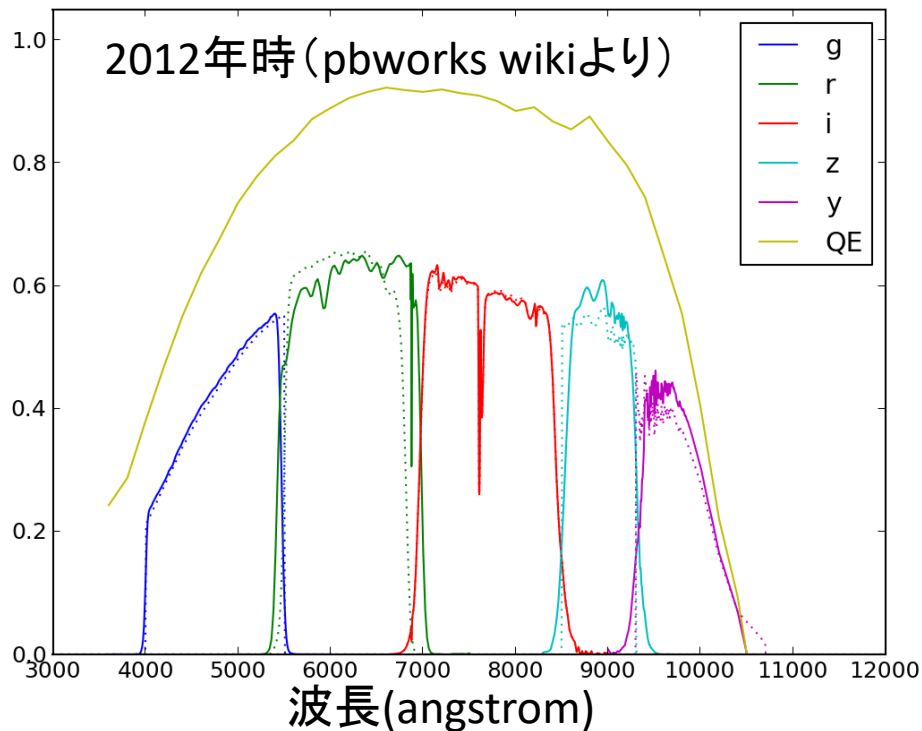
等級較正におけるBand変換

- 較正用星のHSCバンド等級をPS1等級の色の関数として決めている
 - $m_{\text{HSC}}[\text{mag}] = -2.5 \cdot \log_{10}(\text{count} [\text{ADU}]) + a$ (等級原点)
- 狭義のcolor termsについて
 - HSC nativeバンド=実際の観測バンドと仮定しており、両者の実際の差 (b-color) や大気吸収によるバンド変化は取り扱っていない



測光バンドシステム

- 一般に「装置スループット = QE・デュワー・フィルター・レンズ光学系・主鏡」+「大気透過率」を含めたバンドレスポンスで定義されます



SCam:
Johnson-
Cousins
と本物の
Johnson-
Cousins

解析手続きの非常に端折った解説： カタログ作成

- 天体カタログ作成

- カタログとは

- 天体の位置、フラックス、形状などの情報を1天体1行ずつリストアップしたもの
 - 検出 (Detection) と測定 (Measurement) からなる

- 天体検出

- 一般には、あるカーネルで画像にスムージングをかけたうえで、ある条件でスカイのノイズより十分カウントの大きいものを抽出する
 - N-connected pixel法
 - Peak finding 法

- 測定

- 天体検出で見つかった天体ピークの個々に対して以下のような量を測る。
 - 重心、種々のカウント測定、広がり(輝度重み付きの重心からの距離の1次、2次モーメント)、広がりの方向角度

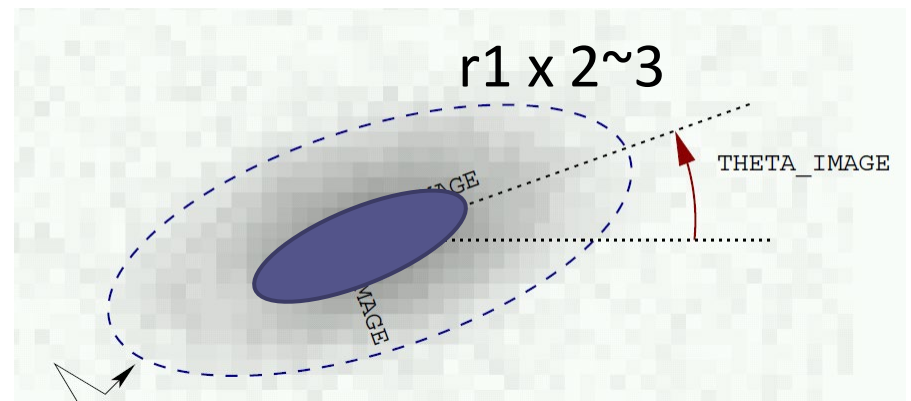
天体プロフィール

- プロファイルの2次モーメント
- CModel = deV.とexp. lawの2成分によるフィット
 - 楕円銀河(バルジ)的 (de Vaucouleurs law; $\frac{1}{4}$ 乗則)
 - $I(r) \propto \exp[-a (r/r_e)^{1/4}]$ (r_e : effective radius)
 - 中心が切り立って裾が広い
 - 円盤・渦巻き銀河(ディスク)的 (exponential law)
 - $I(r) \propto \exp[-b (r/h)^{1/1}]$ (h : scale length)
- 以下は実装されていない
 - Sersicモデル
 - $I(r) \propto \exp[-\nu_n (r/r_e)^{1/n}]$ (h : scale length)
 - Half-light (or effective) radius

Kron測定

- SExtractorのMAG_AUTOで使われていて、すばるの extragalactic分野の研究者がなじんできた測定
- 1次モーメントで楕円アパーチャのサイズを決める
- その中のカウントを

$$r_1 = \frac{\sum rI(r)}{\sum I(r)}$$

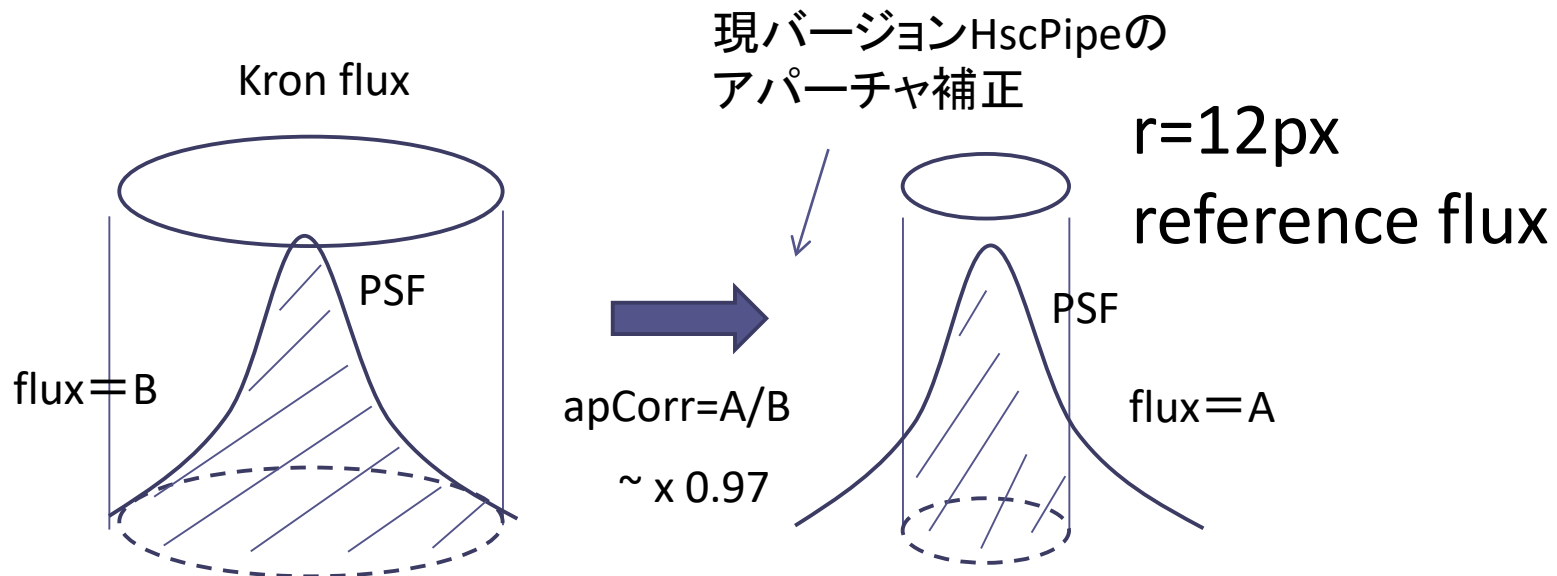


From SExtractor's manual

- hscPipeではCModelの方がよく試験されている。
必要な場合は注意して使うのが良い。

アパーチャ補正

- 各測定アルゴリズム (KronとかCModelとか) でのPSF様天体の測定カウントが $r=12\text{px}$ の aperture flux = 'reference' での測定カウントに一致するようにするための補正因子
- 座標の関数 (2次Chebyshev)
- hscPipe出力のカタログでは, aperture flux以外の測定アルゴリズムに適用されている

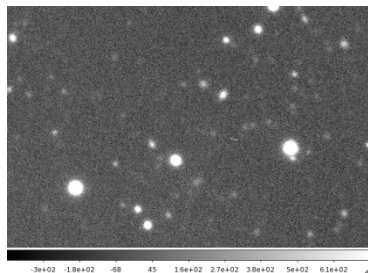


CORRファイルのHDU構成

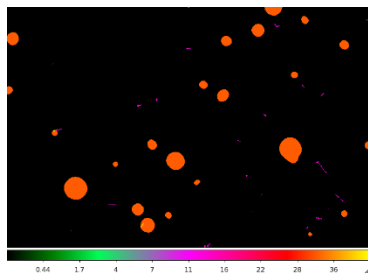
- 全22HDUs (hscPipe7.9.1の場合)
- Primary HDU
 - 基本情報ヘッダのみ
- 2nd HDU
 - Image Pixels (32bit Float) : 圧縮済み
- 3rd HDU
 - Mask (32bit Integer) : 圧縮済み
- 4th HDU
 - Variance (32bit Float) : 圧縮済み
- 5th and larger
 - 付属情報: PSF、アパーチャ補正、較正 (astrometry, photometry)

マスクとフラッグ

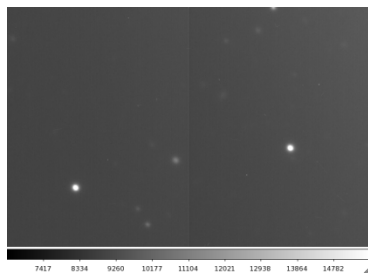
HDU0: Header only
HDU1: Science Image



HDU2: Mask Image



HDU3: Variance Image



Flags

- 検出関係: bad, sat, intrp, cr, edge, detected, suspect, crosstalk
- 測定関係: extendedness, deblend, 各種failure

ビットとマスクイベントの対応は不定
ヘッダのマスクレイヤーを参照のこと

Plane 0 -> BAD
Plane 1 -> SAT
Plane 2 -> INTRP
Plane 3 -> CR
Plane 4 -> EDGE
Plane 5 -> DETECTED
Plane 6 -> DETECTED_NEGATIVE
Plane 7 -> SUSPECT (>50000ADU)
Plane 8 -> NO_DATA
Plane 9 -> UNMASKEDNAN
Plane 10 -> CROSSTALK
Plane 11 -> NOT_DEBLENDED

CCD解析： 出力ファイル一覧

- WCS & magzero付きの解析結果画像
 - corr/CORR-\$(visit)-\$(ccd).fits (FITS Image + BINTABLE)
- スカイ引き情報
 - corr/BKGD-\$(visit)-\$(ccd).fis (FITS Image) 単ショット128px-mesh
 - Corr/skyCorr-\$(visit)-\$(ccd).fits (FITS Image) グローバルスカイ補正
- CCDごとの天体測定カタログ
 - output/SRC-\$(visit)-\$(ccd).fits (FITS BINTABLE)
- Matchlist
 - 検出天体とPanSTARRS(PS)1外部カタログとをクロスIDしたカタログ
 - output/ {SRCMATCH, SRCMATCHFULL}-\$(visit)-\$(ccd).fits (FITS BINTABLE)
- オーバースキャン引き後、フラットフィールド後スナップショット画像
 - thumbs/OSS、flattened

解析手続きの非常に端折った解説： モザイクスタック

- CCD合成作業(モザイクキング・スタッキング)

- 目的

- CCDの間は隙間が空いているので、少しずつ位置をずらしながら取った複数ショットを重ね合わせて隙間を埋め、かつS/N(天体信号のコントラスト)を上げる

1. モザイクキング

- CCD間の相対位置を決める
- CCD間のフラックススケールを決める

2. ワープ&スタッキング

- 目標とする天域を決め、そこでの投影となるように各CCD画像を画像変換する
- フラックスのリスケールを行う
- 重なり合うCCD画像同士をmedianあるいは外れ値除去を含む平均で足し合わせる
 - この際、各CCDに対して一般的にはノイズの大きさを使って重みをつける
 - PSFのモデル情報も合成して保存する

1. モザイクング(2)

- hscPipeにおける留意点
 - ~~1ノード実行。逆行列計算部はOpenBLAS・またはMKLのSMPマルチスレッドにより実行される~~
 - ~~WCSは、TAN-SIP 9次ので各CCDごとにマッピングされる~~
 - ~~photoCalib(Fcr)は、FLUXMAGO(0次)+7次Chebyshevで各CCDごとに決定される~~
 - Distortionによる輝度変化の効果は、モザイクングによるwcs & photoCalib決定で初めて測定され、次のワープ解析で初めて補正される
 - r=12px(やや小さい)で等級原点較正用のカウントが測られている効果は同様にphotoCalibにも引き継がれる
 - 視野端ほど天体はdistortionにより歪んでおり、この効果は視野端の方が強い(1%弱くらい出る可能性がある)
 - FGCMと最新のwarp時にどのように扱われているかは未調査

hscPipeの場合のdistortionによる測光誤差補正

- Mosaic
 - WCSが正確に決まる
 - →場所ごとのピクセルスケールが分かる
 - →fluxに施すべきこの効果の補正量が分かる
 - →この効果以外の測光誤差(主にflatfield誤差)も測定
 - → jointcal_photoCalib へ記録
- Coadd
 - Jointcal_photoCalibを使い, 上両者の測光誤差をcoadd時に補正
- calibrateExposure, calibrateCatalog
 - Jointcal_photoCalibを使い, 上両者の測光誤差をCCDに対して補正
- ※この効果による場所による測光誤差(較正原点の差)
 - 視野中心と視野端で~7%
 - 視野端のCCD面内の場所による測光誤差が0.07—1%くらい

画像の圧縮

- hscPipe 6.7以降、FITSの画像データはすべて圧縮形式
 - Tiled Image Compression -> 詳しくはFITS Standard等参照
- カタログは非圧縮
- 最近のFITS入出力ソフトウェアであれば読める
 - ds9
 - astropy
 - hscPipe (afw)

その他

- Known issues
 - CCDごとの画像・測光結果 (SRC、CORR)をサイエンスに使う場合、モザイク(jointcal)の結果を反映しないと厳密には正しくない。この操作を行うcalibrateExposure, calibrateCatalogの動作が十分試験されていないので、個々のCCD解析結果を使う際には注意が必要
 - Jointcalの測光の問題は明示的に解決されていない
 - SSPのissuesのページを参照するとSSPのデータ解析で分かっている制限・課題が分かる
 - 最近のパイプラインについてはLSSTのフォーラムやチュートリアルも参考になるかもしれない。
 - ほかにもあるかもしれないです