

光赤外撮像観測と そのデータ処理

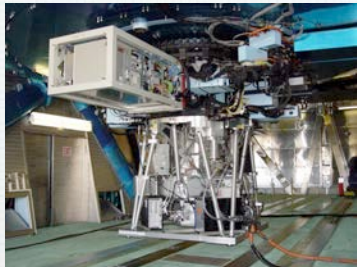
2017-08-03 中島 康

- ◆ こういった画像はデジカメでぱちっと撮るように簡単に出てくるものではない
- ◆ 適切な観測手法とデータ処理を必要とする



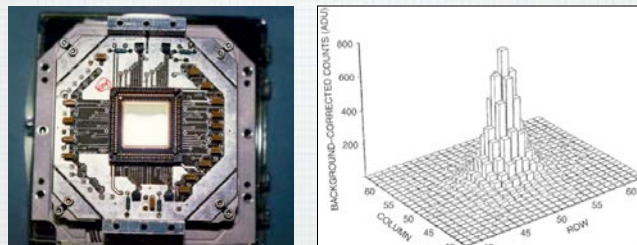
光赤外撮像観測

- ◆ 望遠鏡の焦点に観測装置を置いてデータを取得する

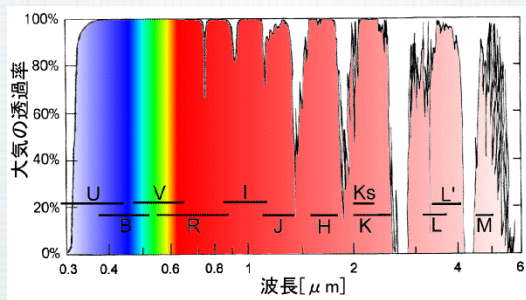


2次元アレイ検出器

- ◆ 観測装置の中で、フィルターを透過した光子を2次元アレイ検出器(CCD, CMOS)の各ピクセルがとらえる。



フィルター (バンド)



可視光: U, B, V, R, I バンド

近赤外: J, H, K, L, M バンド (大気の窓に対応)

観測での積分

integration, exposure

- ◆ アレイ検出器に望遠鏡からの光を当てた状態で、ある時間の間に来た光をカウントすることを**積分する**という。20秒積分とか言う。(カメラの露出)

生データ raw data

- ◆ アレイ検出器から直接出てきたデータのこと
- ◆ 生データには、
 - ◆ 天体以外の信号を含む。除きたい
 - ◆ ピクセル間の感度ムラ、光学系の透過ムラの影響を受ける。補正したい
 - ◆ 光子の個数をADUに変換している。Jyあるいはerg/s/cm²/Hzあるいは等級が知りたい

生データ取得 + 一次処理に必要なデータ取得

天体以外の信号の除去

感度と光学系のムラの補正

一次処理

一次処理済みデータ

例) 点光源の測光 (+ 標準星データ取得)

測光値の較正

データ処理のために観測時に
やっておくことがある

生データの特性は観測装置に依存

同じように見える二次元アレイ検出器だが、
観測装置、波長、条件(地上かスペースか)
で必要な手順(観測、データ処理)は異なる

1. まずはシンプルな可視光での手順

2. 少し複雑な近赤外での手順

講習ではこちらを使った実習

1. まずはシンプルな

可視光

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

- ◆ flat(x,y) : ピクセル間の感度むら, 光学系の透過むら
 - ◆ object(x,y)=const. の光がやってきたとしても flat(x,y) × const. が検出される。
- ◆ 近年のCCDではdarkは無視できるレベル
- ◆ biasは、raw(x, y)のオーバースキャン領域から推測 (x方向への依存は無しと仮定し)

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

flat(x, y)

object(x, y)

1.1	0.9	0.9	×	200	204	201	=	220	183	181
1.0	1.2	0.8		198	400	202		198	480	162
1.1	0.7	1.1		205	199	200		226	139	220

1. まずはシンプルな

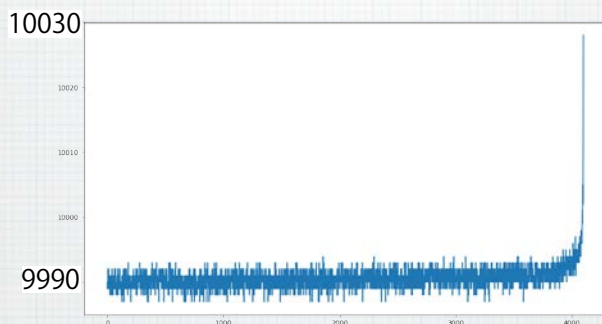
可視光

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

- ◆ $\text{flat}(x, y)$: ピクセル間の感度むら, 光学系の透過むら
 - ◆ $\text{object}(x, y) = \text{const.}$ の光がやってきたとしても $\text{flat}(x, y) \times \text{const.}$ が検出される。
- ◆ 近年のCCDではdarkは無視できるレベル
- ◆ biasは、 $\text{raw}(x, y)$ のオーバースキャン領域から推測 (x方向への依存は無しと仮定し)



Y方向にも大部分で一様で、端のほうで0.数% 増加



可視光

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

- ◆ $\text{flat}(x, y)$: ピクセル間の感度むら, 光学系の透過むら
 - ◆ $\text{object}(x, y) = \text{constant}$ の光がやってきたとしても $\text{flat}(x, y) \times \text{constant}$ が検出される。
- ◆ 近年のCCDではdarkは無視できるレベル
- ◆ biasは、 $\text{raw}(x, y)$ のオーバースキャン領域から推測 (x方向への依存は無しと仮定し)

他のデータ取得不要

フラット

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

$\text{object}(x, y) = \text{constant}$ (一様光)を観測してやれば、

$$\text{flat}(x, y) = \frac{\text{raw}(x, y) - \text{bias}(x, y)}{\text{constant}}$$

$\text{constant} = \text{median}(\text{raw}(x, y) - \text{bias}(x, y))$
規格化

ドームフラット

望遠鏡ドームを閉じて、フラット板に照明をあてて撮影

望遠鏡の筒のすぐ近く

-> ピント合わず像がぼけまくり

-> ムラは無い(かも) 一様光

何枚かの平均から、尤もらしい $\text{flat}(x, y)$ を作成する

スカイフラット

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

で、

$$\text{object}(x, y) = \text{stars}(x, y) + \text{constant}(\text{上空の散乱光})$$

としたとき、長時間積分を行い、

$$\text{stars}(x, y) \ll \text{constant}$$

となる場合には、一様光を観測したとみなせる。

観測時 ドームフラット + 天体 + 標準星

1次処理 ドームフラットデータから $\text{flat}(x, y)$

天体生データから $\text{bias}(x, y)$

$$\text{object}(x, y) = \frac{\text{raw}(x, y) - \text{bias}(x, y)}{\text{flat}(x, y)}$$

1次処理済みデータができた



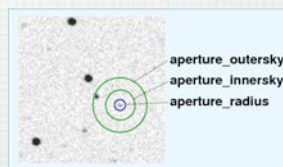
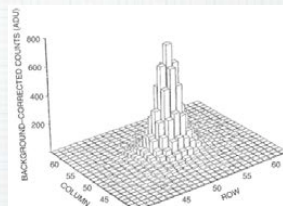
星の測光

標準星による測光値の校正

星の測光

アパーチャ測光

- 星を含む円内のカウントを総計する。
- 明るさの分かっている星のカウントと比べて、明るさを求める。



標準星の観測が必要

観測時 ドームフラット + 天体 + 標準星

1次処理 ドームフラットデータから $\text{flat}(x, y)$

天体生データから $\text{bias}(x, y)$

$$\text{object}(x, y) = \frac{\text{raw}(x, y) - \text{bias}(x, y)}{\text{flat}(x, y)}$$

1次処理済みデータができた



星の測光

標準星による測光値の校正

可視光

$$\text{raw}(x, y) = \text{flat}(x, y) \times \text{object}(x, y) + \text{dark}(x, y) + \text{bias}(x, y)$$

近赤外 (地上)

$$\text{raw}(x, y) = \text{flat}(x, y) \times [\text{object}(x, y) + \text{skybias}(x, y)] + \text{dark}(x, y)$$

- 近赤外検出器ではdarkは無視できない
- skybias: 熱放射などによる時間変動する非一様なパターン

ダーク

- $\text{dark}(x, y)$: 検出器に光があたらずとも積分時間に応じて載ってくる成分。
- シャッターを閉じて、天体と同じ積分時間でダーク用データを何枚か撮る。観測前や後に撮る
- 何枚かの平均から、尤もらしいダーク補正フレームを作成

フラット

$$\text{raw}(x, y) = \text{flat}(x, y) \times [\text{object}(x, y) + \text{skybias}(x, y)] + \text{dark}(x, y)$$

赤外では熱放射などによるskybiasが加わるため、
一様な光を異なる明るさで観測する必要がある

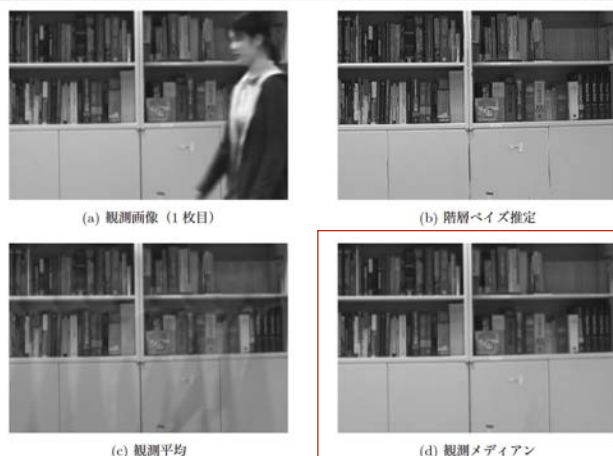
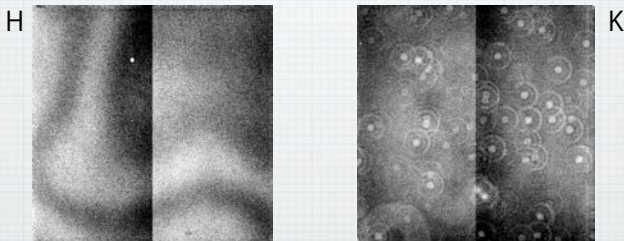
$$\begin{aligned} \text{raw1}(x, y) &= \text{flat}(x, y) \times [\text{Const1} + \text{skybias}(x, y)] + \text{dark}(x, y) \\ \text{raw2}(x, y) &= \text{flat}(x, y) \times [\text{Const2} + \text{skybias}(x, y)] + \text{dark}(x, y) \end{aligned}$$

$$\text{flat}(x, y) = \frac{\text{raw1}(x, y) - \text{raw2}(x, y)}{\text{Const1} - \text{Const2}}$$

- ◆ **ドームフラット** ドームを閉じて、フラット板に照明を **ON**と**OFF**の状態撮影。
- ◆ **トワイライトフラット** 薄明時に空の明るさがどんどん変わって行くのを観測。
- ◆ **スカイフラット** スカイバイアスが無視できる場合
- ◆ 検出器の感度ムラや光学系の透過ムラなんていうのは、そう変わったりするもんでもないので、何ヶ月かにわたる蓄積のフラットを使わせてもらってもよい。

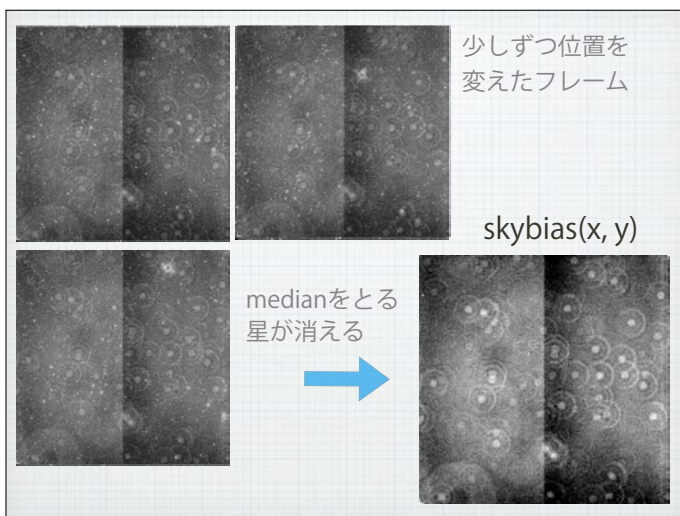
スカイバイアス

- ◆ 主にKバンドより長い波長で熱的な放射成分(300K)
- ◆ 主にHバンドでOH夜光の輝線成分によるフリンジパターン(newton ring)
- ◆ HAWAII array 特有のリセットアノマリースロープ



skybias(x, y)を作る

- ◆ 天体を観測する近い時間のうちに
- ◆ その天体に近い方角で星が込み合っていない視野を
 - ◆ (条件を満たしていれば自分自身でもよい:セルフスカイ)
- ◆ 少しずつ望遠鏡の向きを変えて
- ◆ その天体と同じ積分時間で
- ◆ 10枚くらい撮る。
- ◆ 全てのピクセルが、10枚のうちほとんどで星のない空を見るとすると、これらの画像の平均(median)をとることで星が消えて、skybiasだけがのこる



観測時 ドームフラット + 天体 + 標準星

1次処理 ドームフラットデータから flat(x, y)
ダークデータから dark(x, y)
スカイ視野データから skybias(x, y)

$$\text{object}(x, y) = \frac{\text{raw}(x, y) - \text{dark}(x, y)}{\text{flat}(x, y)} - \text{skybias}(x, y)$$

1次処理済みデータができた？

近赤外(地上)の天体そのものの観測

(1) 常温では何でも光る

- ◆ 望遠鏡の主鏡や大気は常温(~300K)なので赤外線でも光る。背景のカウント数がすぐに上昇。
- ◆ 各ピクセルに貯めれるカウント数には上限がある。(Saturation: サチる) 1枚で長時間積分できない。

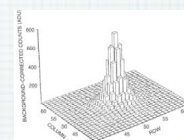
20秒積分 x 10枚 -> 200秒積分のようにする。

(2) bad pixel

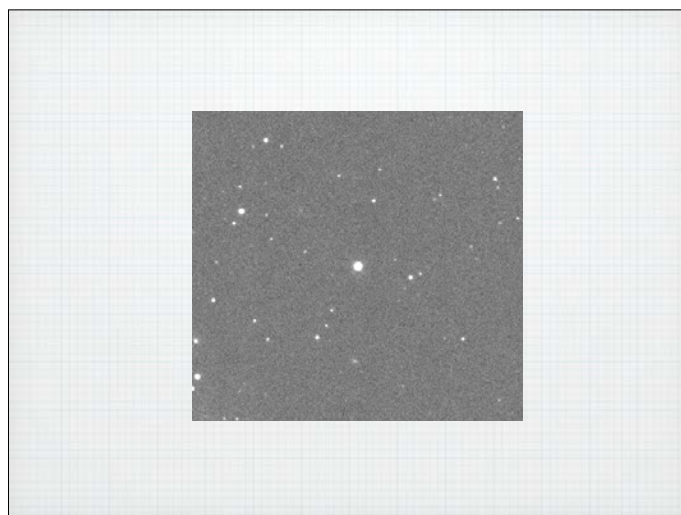
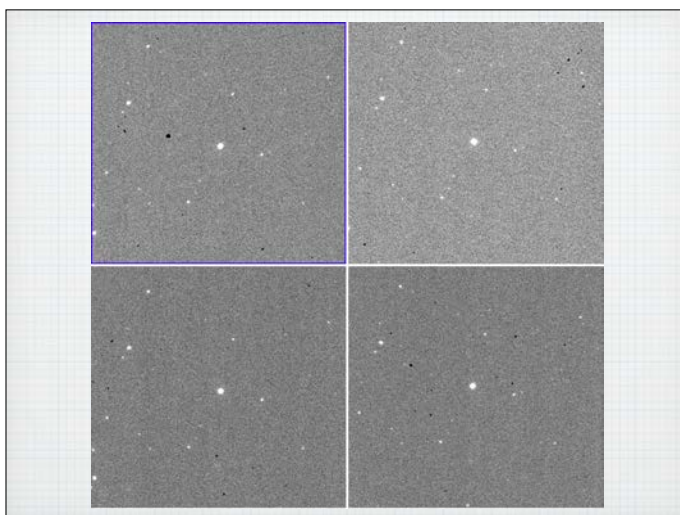
- ◆ 赤外アレイ検出器には**バッドピクセル**が多い
- ◆ 望遠鏡の向きを少しずつ変えて、次々と撮っていく。

ディザリング(dithering)

- ◆ 視野の広さよりは十分に小さく
- ◆ 星の広がり(fwhm)よりは大きく



- ◆ 同じ空の範囲がバッドピクセルに常に落ちないようにする。星が重なるように全ての画像を重ねて平均をとる(+clipping)と最終的にバッドピクセルが消える。



位置合わせと重ね合わせ



生データ取得 + 一次処理に必要なデータ取得

天体以外の信号の除去

感度と光学系のムラの補正

一次処理

一次処理済みデータ

例) 点光源の測光 (+ 標準星データ取得)

測光値の較正

データ処理のために観測時に
やっておくことがある

コンピュータでのデータ処理

データ形式: FITSファイル

データ処理アプリ: IRAF

プログラミング言語: Python

FITSファイル Flexible Image Transport System

標準的な天文データフォーマット (NASAとIAUが支持)

拡張子 .fits

ヘッダ部(ASCIIテキスト)とデータ部(バイナリ)から成る

```
SIMPLE = T / Fits standard
BITPIX = -32 / Bits per pixel
NAXIS = 2 / Number of axes
NAXIS1 = 1557 / Axis length
NAXIS2 = 1558 / Axis length
EXTEND = T / File may contain extensions
OBJECT = 'p9182n3std' / Object Name
FILTER = 'H' / Filter Name
EXPOS = 10.0 / Exposure time(s)
DATE_UTC = '2001/10/29' / yyyyymmdd
TIME_UTC = '19:09:36.3' / hhmmss
DATE_JD = '2001/10/29' / yyyyymmdd
TIME_JD = '21:09:36.3' / hhmmss
RA = '20:41:05.1' / hhmmss
DEC = '-05:03:42.9' / ddmmss
ALDOSS = '1.375988' / ALDOSS
NCOMBINE = 9
COMBINE = 'average'
CONJUGATE = '1533:1024,534:1024'
REJECT = '5.0-sigclip MAD'
END
```

ヘッダ部

データ部

IRAF Image Reduction and Analysis Facility

- ◆ アメリカのNOAO (国立光学天文台)が作成 (1986~)
- ◆ 光赤外データ処理・解析のデファクトスタンダード
- ◆ 画像演算、測光などのタスク(プログラム)の集合
- ◆ 長期間、多くの人によって使われてきた「枯れたシステム」 --- バグが淘汰されている
- ◆ IRAFのコマンドラインから対話的データ処理
- ◆ 独自のプログラム言語 IRAF-CL

Python

- ◆ 今、最も広く使われているコンピュータ言語のひとつ
- ◆ PyRAFが「仲介」することでPythonからIRAFのタスクを利用できる！
- ◆ 数多くのライブラリ — 数値計算、統計、可視化、機械学習、、、、
- ◆ 多くの教科書、ウェブ上の情報

IEEEによるプログラム言語総合ランキング2017

Language Rank	Types	Spectrum Ranking
1. Python	🌐 🖥️ 📱	100.0
2. C	📱 🖥️ 📄	100.0
3. Java	🌐 📱 🖥️	99.4
4. C++	📱 🖥️ 📄	96.9
5. C#	🌐 📱 🖥️	88.6
6. R	🖥️	88.1
7. JavaScript	🌐 📱	85.3
8. PHP	🌐	81.1
9. Go	🌐 🖥️	75.7
10. Swift	📱 🖥️	74.3

就職に有利なランキング2017

Language Rank	Types	Jobs Ranking
1. Java	🌐 📱 🖥️	100.0
2. C	📱 🖥️ 📄	99.4
3. Python	🌐 🖥️	99.3
4. C++	📱 🖥️ 📄	92.2
5. JavaScript	🌐 📱	89.9
6. C#	🌐 📱 🖥️	86.4
7. PHP	🌐	80.5
8. HTML	🌐	79.7
9. Ruby	🌐 🖥️	76.6
10. Swift	📱 🖥️	76.4

Python

- ◆ 今、最も広く使われているコンピュータ言語のひとつ
- ◆ PyRAFが「仲介」することでPythonからIRAFのタスクを利用できる！
- ◆ 数多くのライブラリ — 数値計算、統計、可視化、機械学習、、、、
- ◆ 多くの教科書、ウェブ上の情報

one more thing

実験ノート(研究ノート)をとろう！

Jupyter notebookがひとつのソリューション