

低ランク行列近似による Tomo-e Gozen動画データの圧縮

森井幹雄 統計数理研究所

池田思朗(統数研)、酒向重行(東大)、大澤亮(東大)

Morii, Ikeda, Sako & Ohsawa (2017) ApJ, 835, 1

装置の概要

▼ 東京大学木曽観測所 超広視野高速CMOSカメラ Tomo-e Gozen

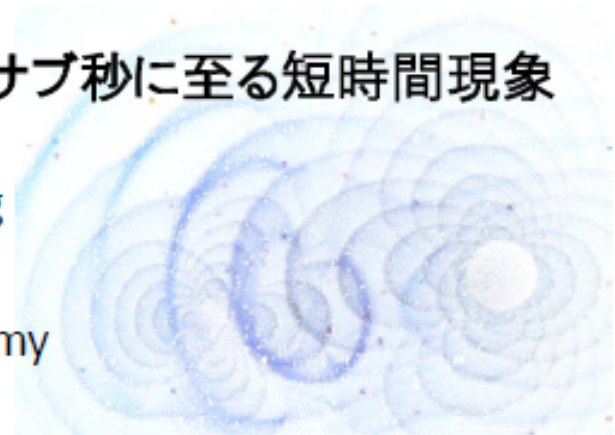
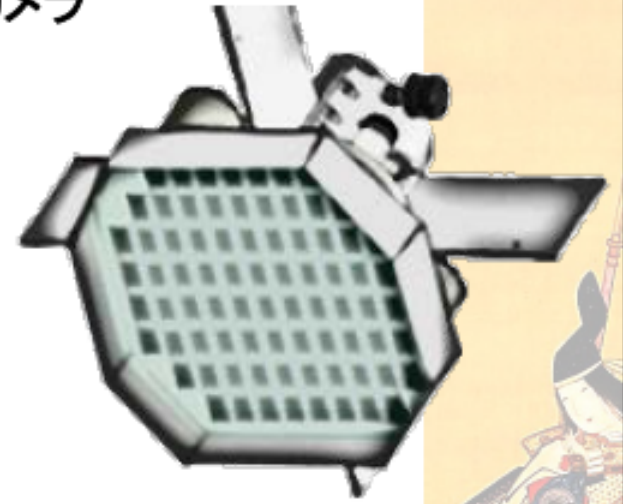
- ❑ Field of view : 20 deg² in ϕ 9 deg
- ❑ Sensor: 84 CMOS chips
- ❑ Frame rate : 2 frames/sec (max)
- ❑ Commissioning : 2017

一晩で30 TB程度

▼ 観測戦略

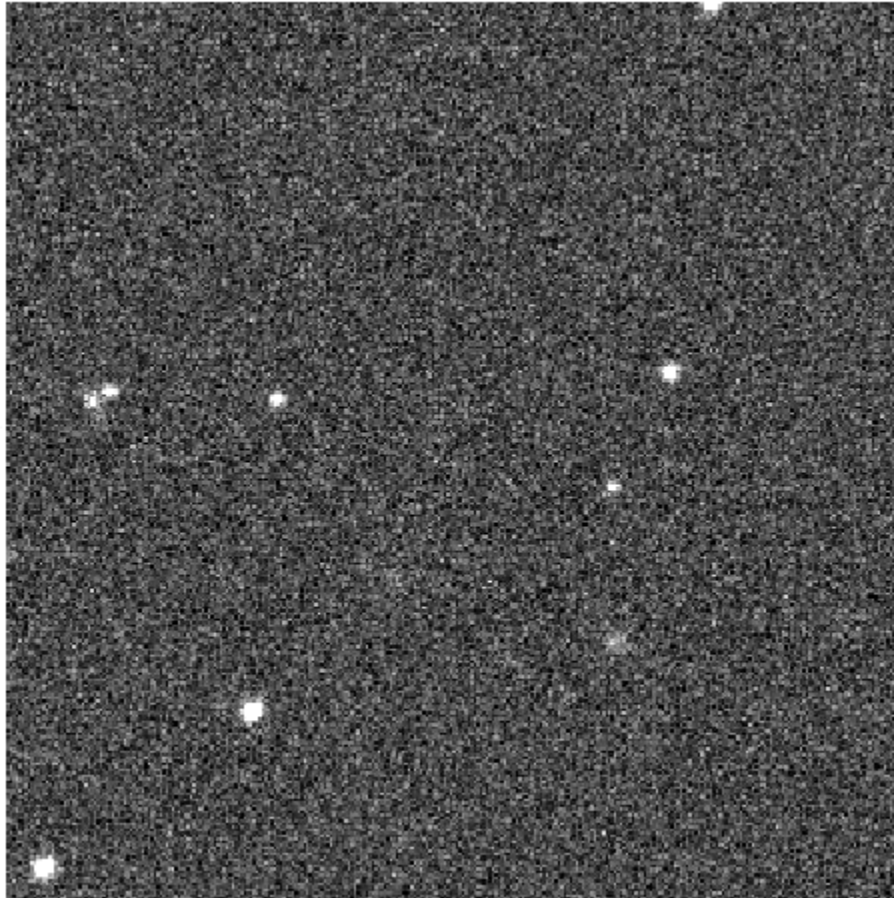
ターゲット → 稀で貴重な突発現象、サブ秒に至る短時間現象

- ❑ **1-hour**-cadence **all-sky** monitoring
- ❑ **20-fps** wide-field monitoring
- ❑ Synergy with **high-energy** astronomy
- ❑ **Near** and **interior** Earth objects



実際に撮られた動画

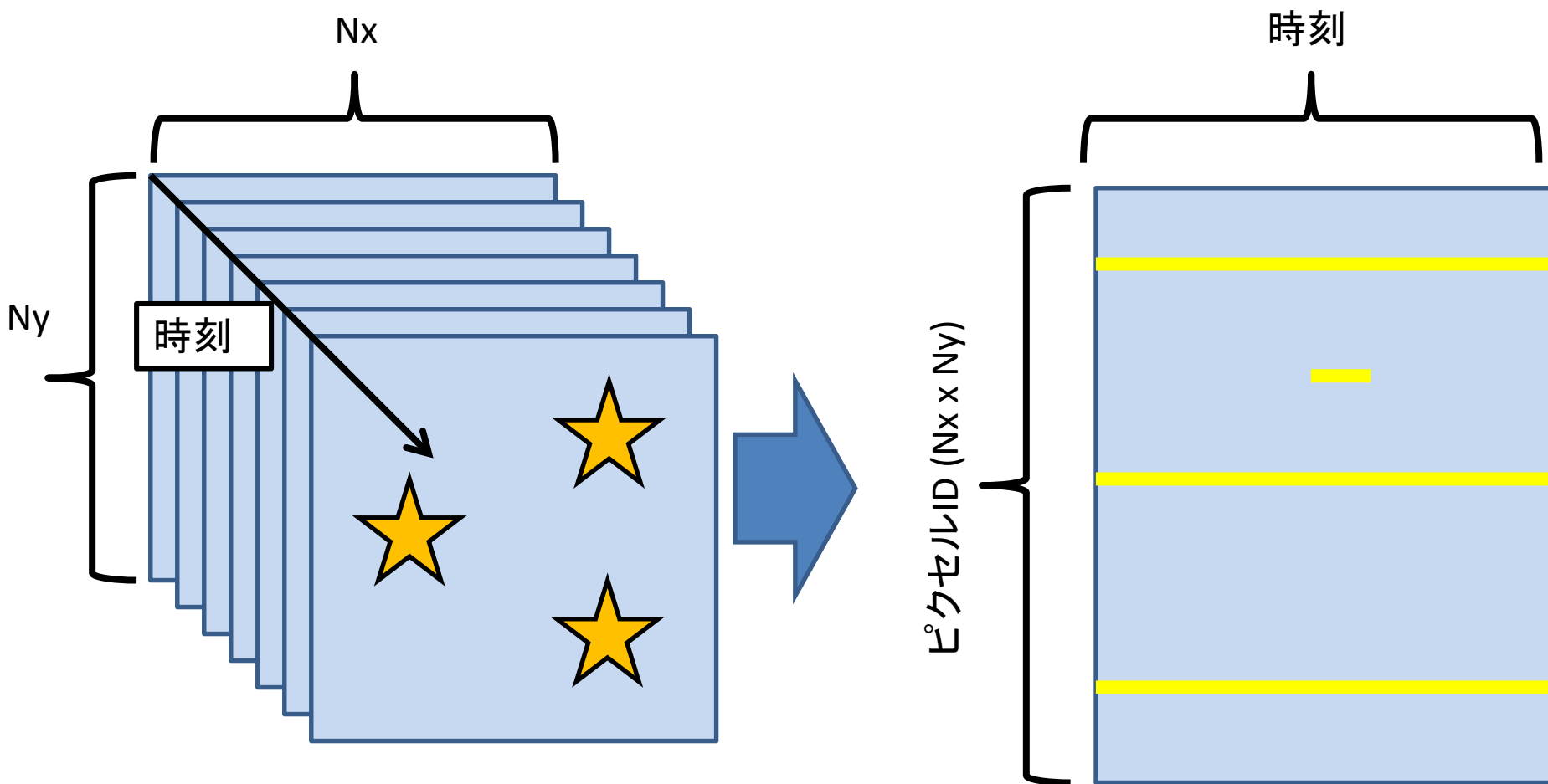
Tomo-e Prototype model で得られた動画の一部 (2 Hz)



酒向さん、大澤さん
より提供して頂いた。

この中から効率的
に突発天体を探し
たい。

3Dデータ(動画)をMatrixで表現



行列分解 (Robust PCA)

- E. J. Candes, X. Li, Y. Ma & J. Wright (2009)
“Robust Principal Component Analysis?”, Journal of the ACM (submitted)

$$\begin{array}{ll} \text{minimize} & \|L\|_* + \lambda \|S\|_1 \\ \text{subject to} & L + S = M \end{array}$$

データ(M) を、
Low Rank Matrix (L) と
Sparse Matrix (S) の和
に分離する。

第1項: Low Rank Matrix の
Nuclear Norm (SVD で得られ
る特異値の和)

第2項: Sparse 行列の l_1 ノルム

パラメータ λ を調整して、
Sparseness を調整する。

行列分解 (GoDec)

- T. Zhou & D. Tao (2011)

“GoDec: Randomized Low-rank & Sparse Matrix Decomposition in Noisy Case”, Proc. of ICML

$$X = L + S + G$$

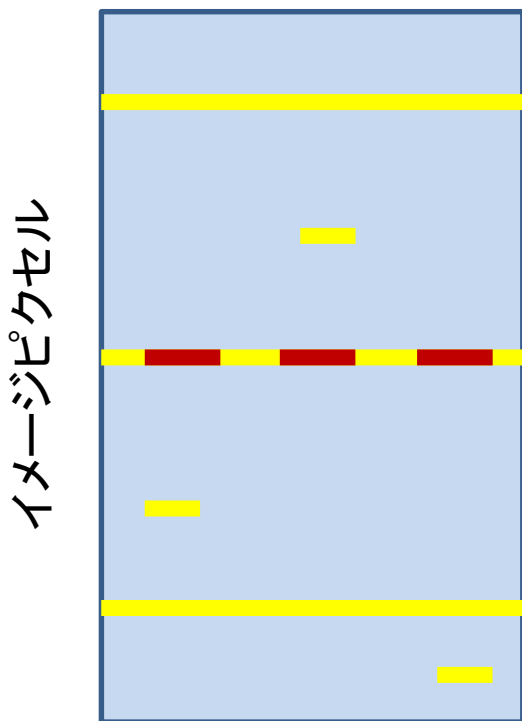
$$\begin{aligned} \min_{L, S} \quad & \|X - L - S\|_F^2 \\ \text{s.t.} \quad & \text{rank}(L) \leq r, \\ & \text{card}(S) \leq k. \end{aligned}$$

X: データを、
L: Low Rank Matrix
S: Sparse Matrix
G: Noise Matrix
の3つの和に分離する

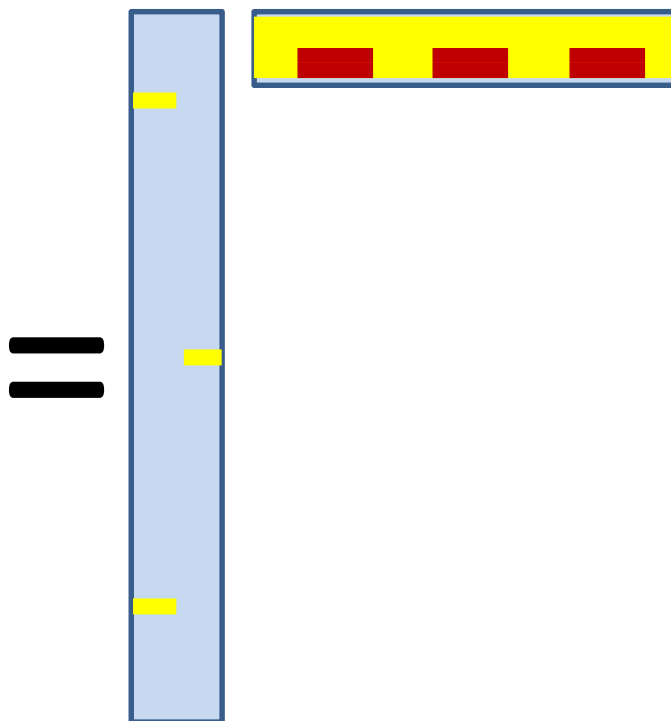
Rank (r)の値と、
sparseness (k)を設定する。

行列分解

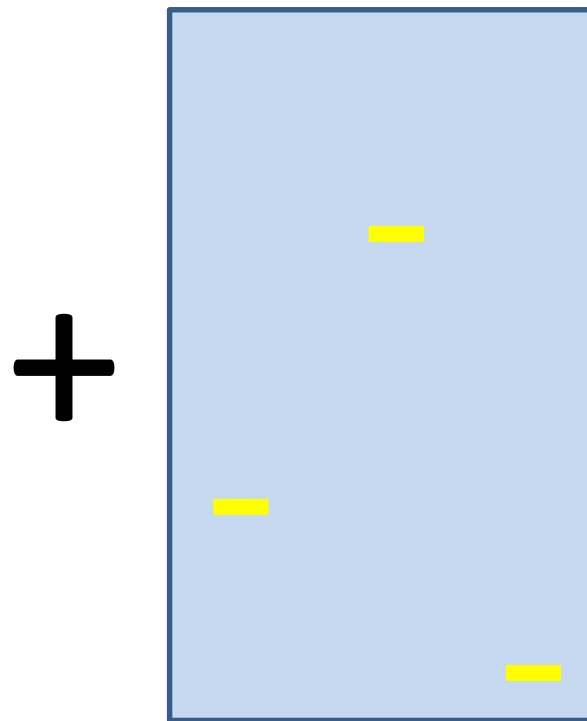
元の行列



Low Rank 行列



Sparse 行列



監視カメラの映像のデモ

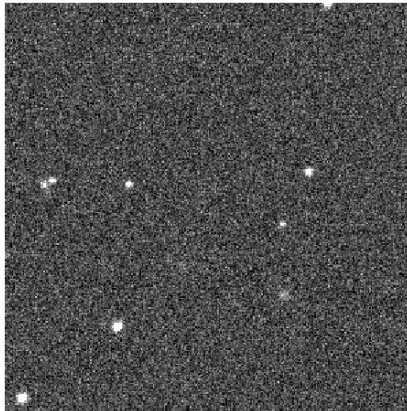
Youtube に在る監視カメラの映像から
動いている人々を抽出するデモ動画

GoDec and Robust PCA for background
modeling on Hall video.mp4

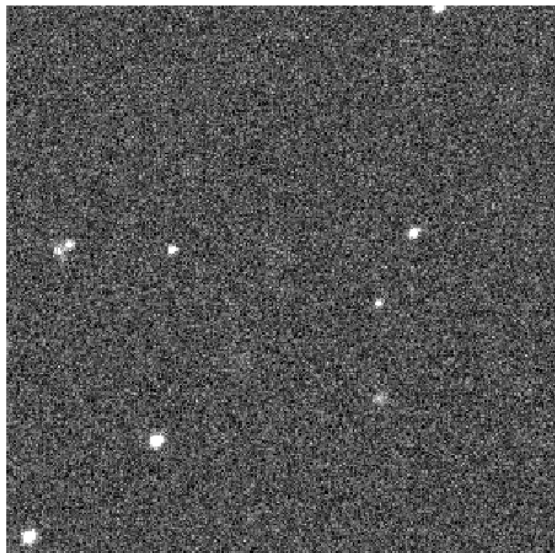
<https://www.youtube.com/watch?v=BTrbow8u4Cw>

行列分解の結果(GoDec)

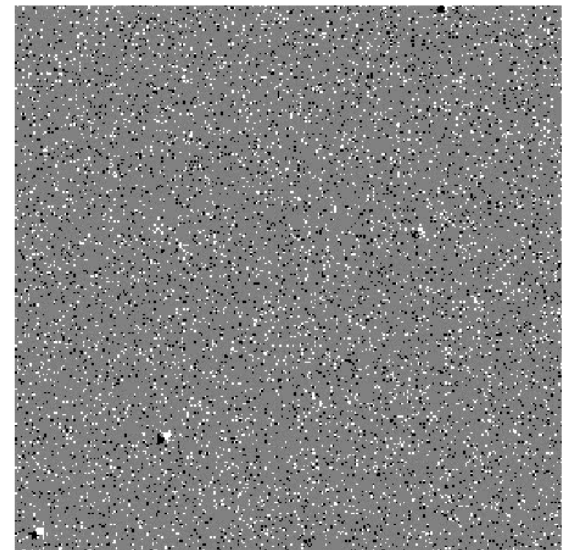
分解前



Low Rank



Sparse

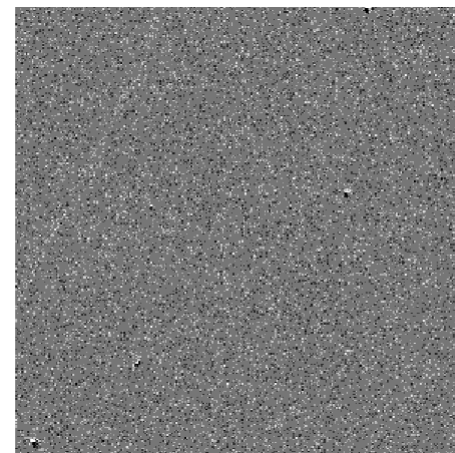
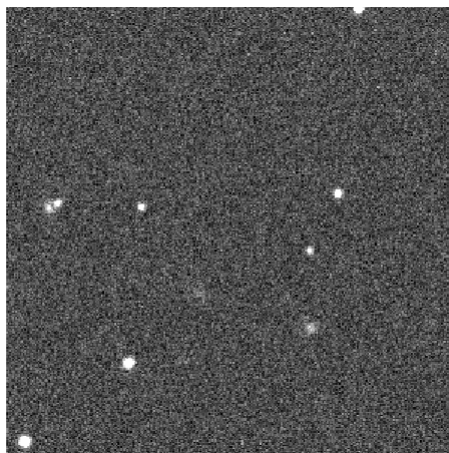
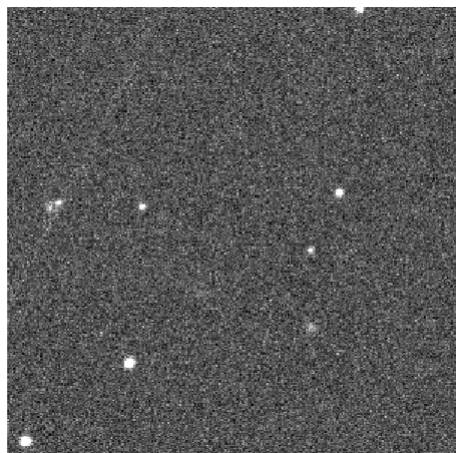
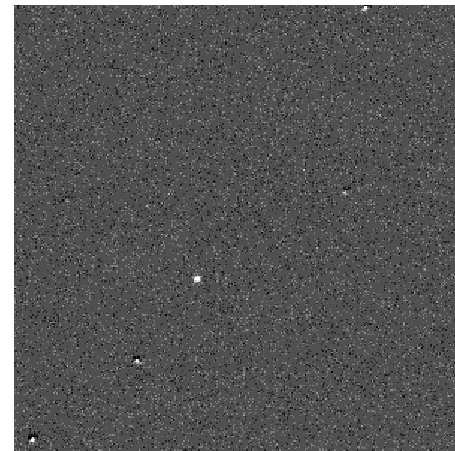
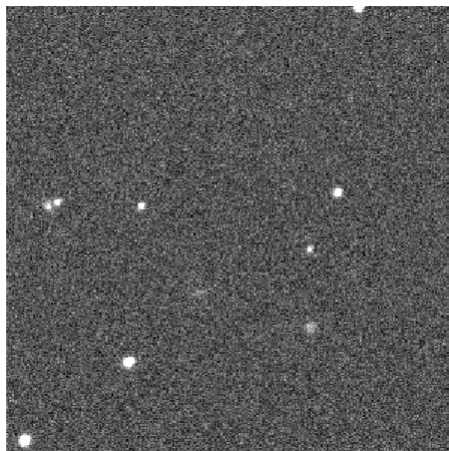
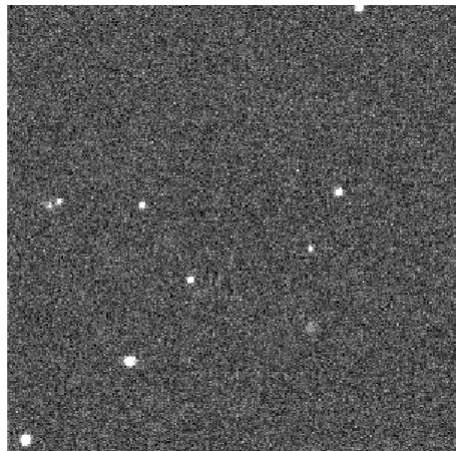


行列分解の結果(GoDec)

元画像

Low Rank

Sparse



突発天体?

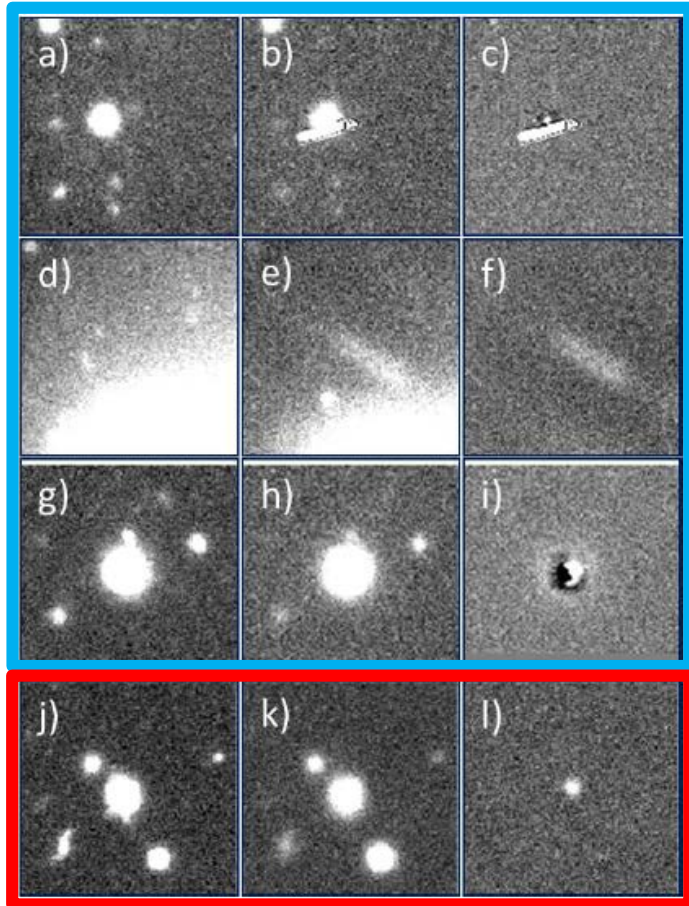
流星

動画観測データから突発天体を探索する方法

- Low-rank 行列近似 (GoDecなど) で行列分解を行い、Sparse 動画から突発現象だけを抜き出す。
- 定常成分は、Low Rank 動画に含まれるため、データを圧縮できる。
- 明るい天体の場合、星の揺らぎ (PSF の変化) のために、Sparse 画像に混入があるが、低解像度の動画で行列分解を行う。
- あるいは、PSF の形状に関する特徴量に対して、機械学習を用いて、本物と偽物の判別を行う。

PSFの形状を用いた突発天体の選出

previous new diff



Subaru望遠鏡の Hyper Suprime-Cam のデータに対して、機械学習を用いた選出を行い、実証済み。

Morii et al. 2016 PASJ, 68, 104

“Machine-learning Selection of Optical Transients in Subaru/Hyper Suprime-Cam Survey”

2015年8月19日の観測

406,121 天体



13 個の特徴量を抽出



Machine Learning
機械3つの多数決

AUC Boosting,
Random Forest,
Deep Neural Net



1,318 天体



目視で確認



ATELに報告(10 天体)

ATEL報告までを一日以内に行うことに成功している。

AUC boosting のソースコード

- ツールの置き場

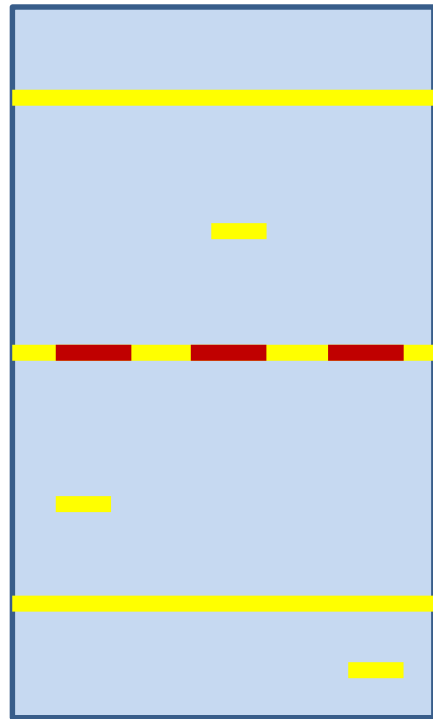
<https://github.com/moriiism/hsc>

- インターフェース部分がきれいではないですが、とりあえず。

データ圧縮

元の行列

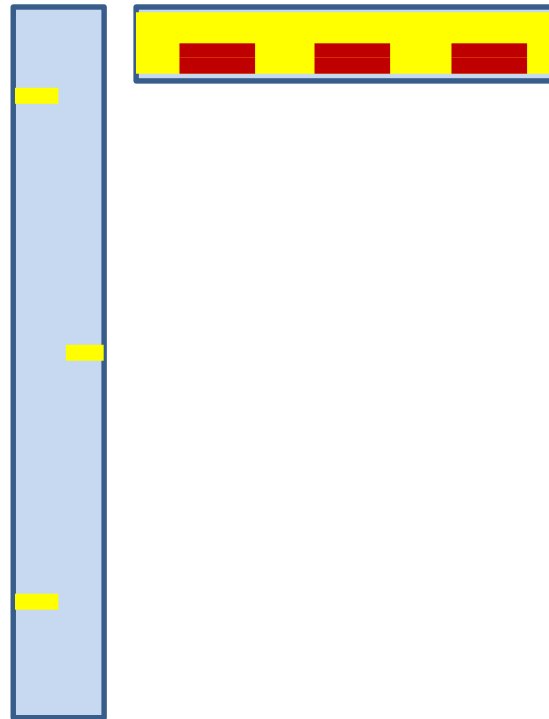
イメージピクセル (10の6乗)



時刻(10の2乗)

=

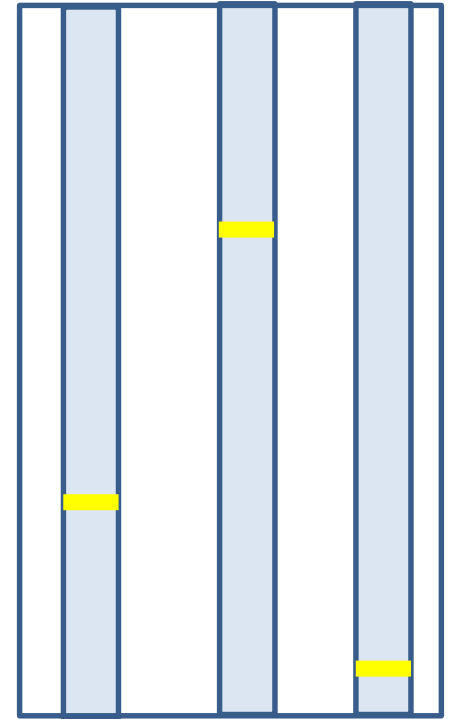
Low Rank 行列



Rank (10)

+

Sparse 行列



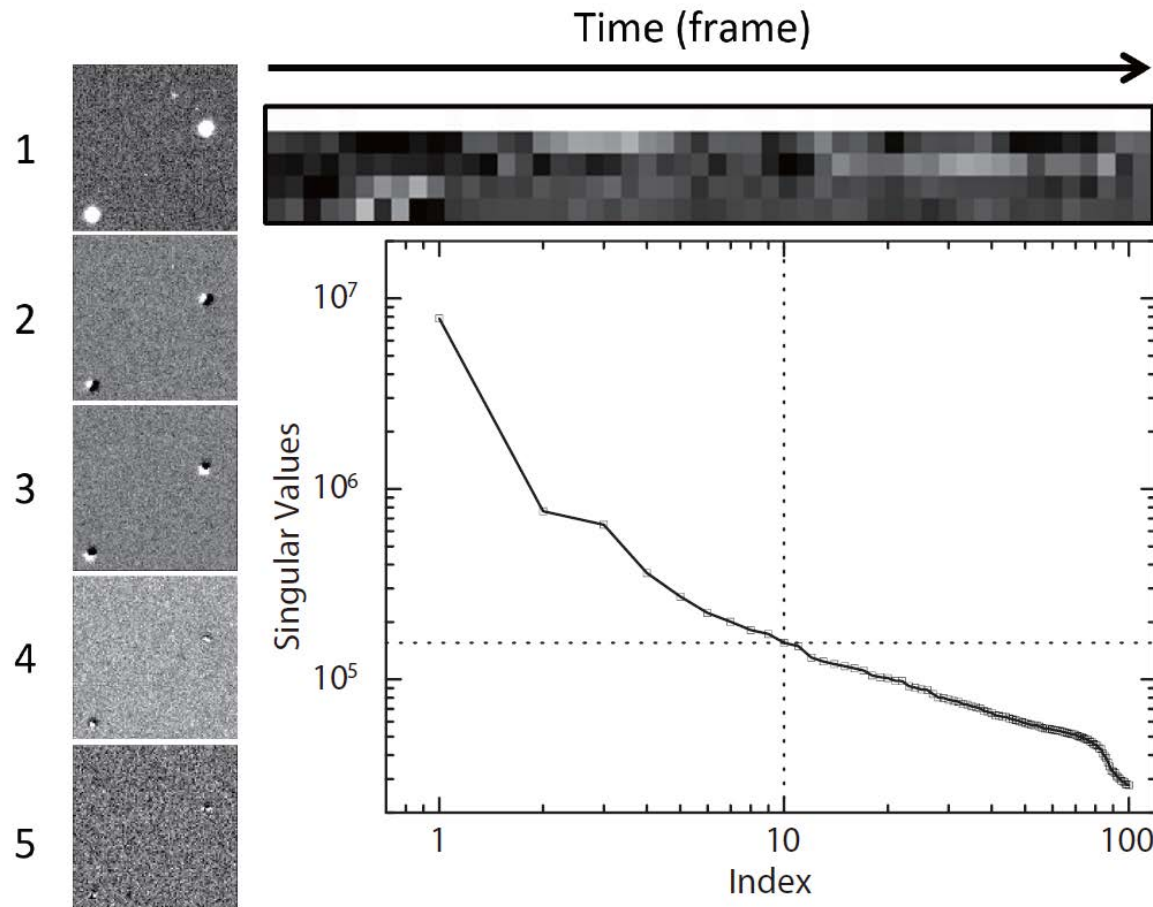
10

$$10^6 \times 10^2 = 10^8 \rightarrow$$

$$10^6 \times 10 + 10^2 \times 10 + 10^6 \times 10 = 10^7$$

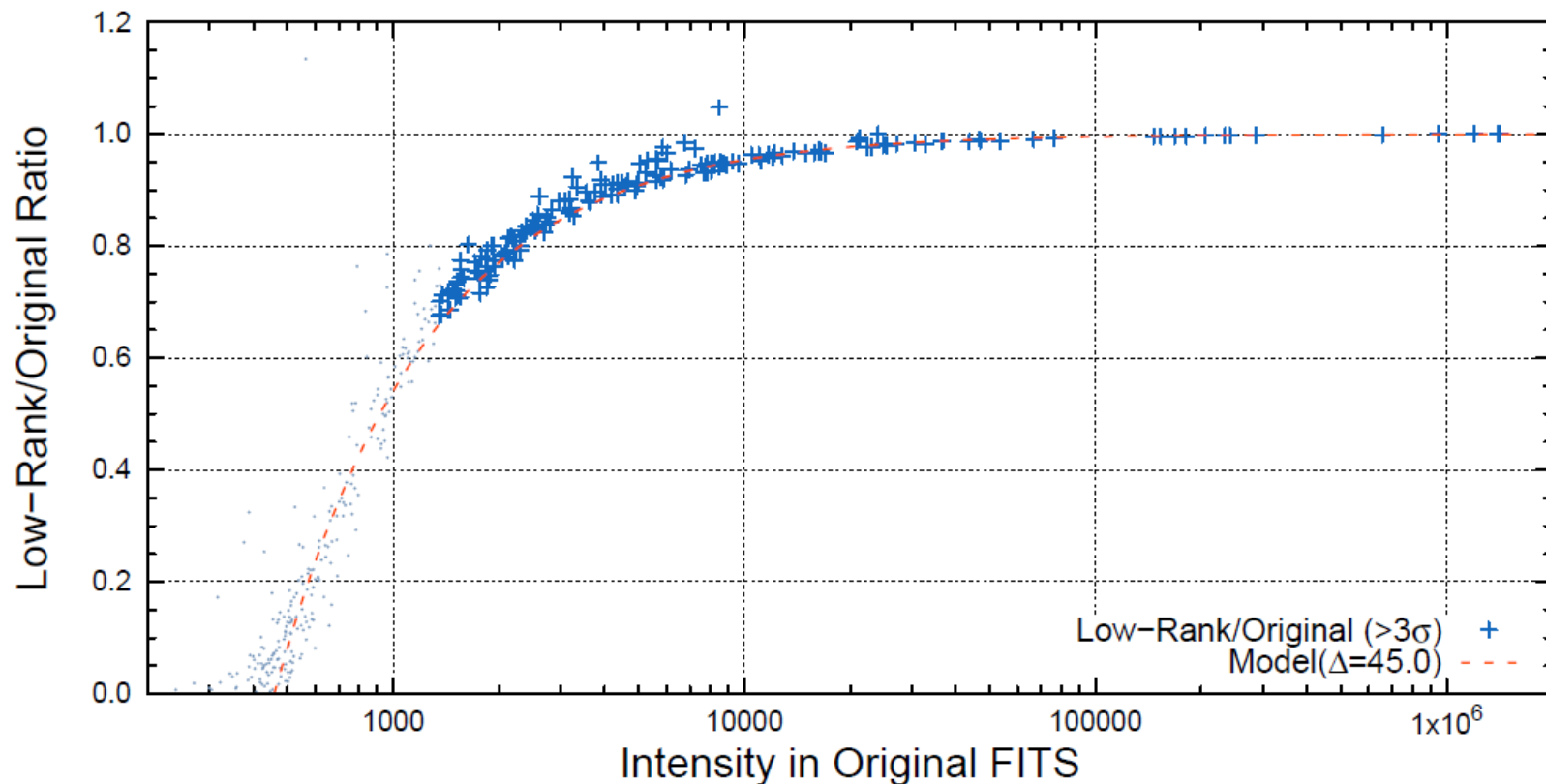
10分の1くらいに
データを減らせる。

Low Rank の動画をSVDする



Rank = 10 程度で十分元動画を再現できる。

点源の明るさの復元



Aperture Size が小さくなると考えて補正すると、うまく復元できることが分かった。
補正式は論文をご覧ください。

処理速度の問題

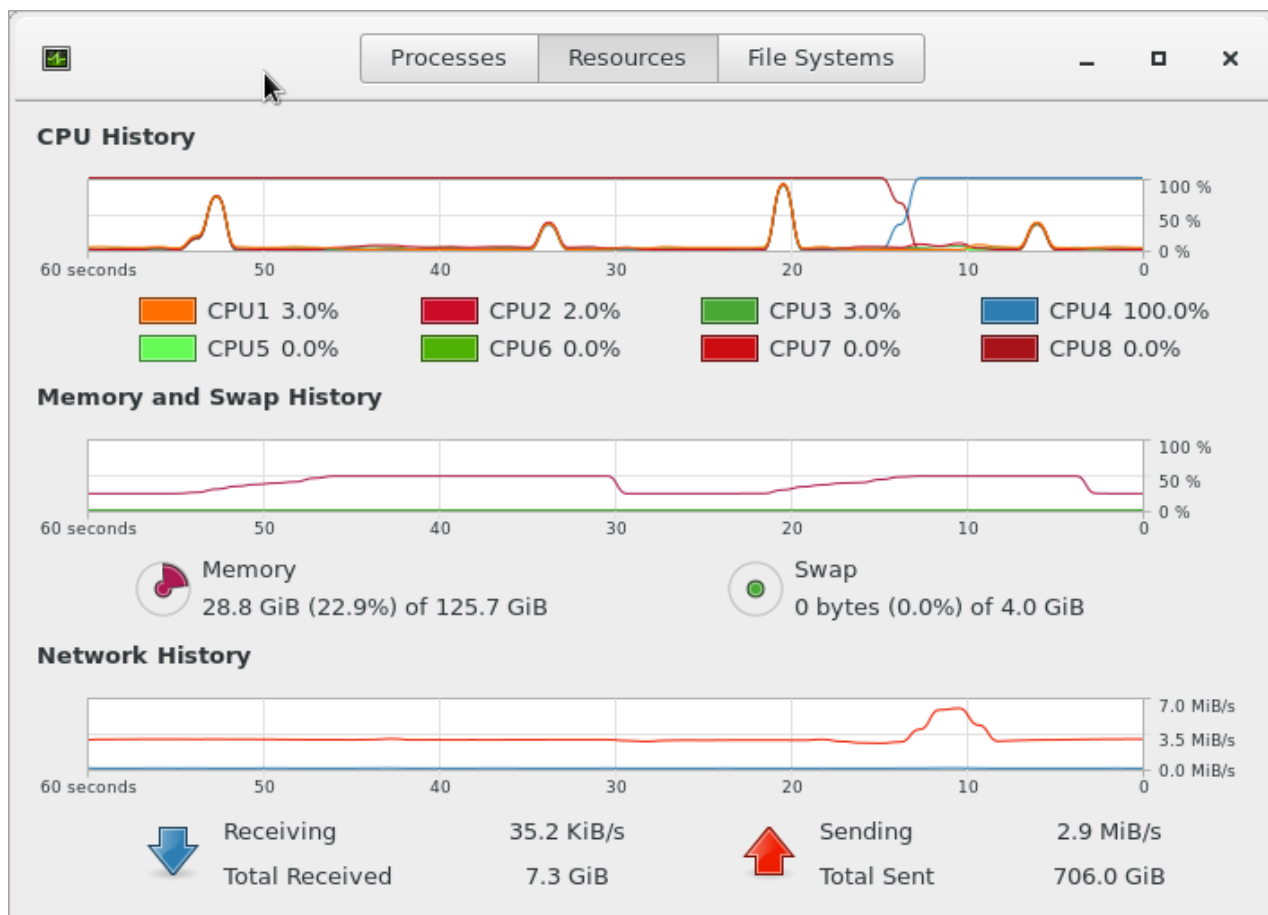
- 2Hzでデータを取った場合、1 chip (1000 x 2000 (pixel)) x 400 (frame)を観測時間(200秒)以内に処理しないといけない。
 - さもないと、データが貯まる一方で破綻する。
- 高速化の工夫
 - SVD を random projection による Range Finder を用いて高速に行う (Halko et al. 2011) 。
 - 線形代数ライブラリ: OpenBLAS を使用
 - Quick Sort -> quick select
- 320秒で処理できるようになった。
 - 観測は、夜だけなのに対して、処理は24時間できるので、一応破綻しない。

本格的な観測データへの適用

- 2016年4月11日、14日の観測で得られた地球の影方向の監視データ
- 4/11 (1.7 TB), 4/14 (1.8 TB).
- 1 file: 2008 x 1136 (pixel) x 400 (frame)
- Rank = 10, Sparsity = 10%, Iteration = 10.
- 処理マシンのスペック
 - Linux CentOS 7.2 (64 bit)
 - CPU: Intel Xeon E5-1630 v3 (3.70GHz)を1個使用。
 - 8コア, Cache: 10 MB
 - Memory: 128 GB

結果

- 3.5 TB の処理に、6.9 日間かった(1ジョブ投入)。
- ピーク時メモリ消費量: 59GB



ほとんど、1コアだけで処理しているが、ときどき、8コア全部で処理している(Open BLASで計算している時)。

メモリ容量が厳しかったので、1つのジョブだけで処理した。
もし、メモリが512 GB搭載されていたら、8つジョブを投入して処理できる。

改善案

- Iteration 回数を10回にしたが、実は半分の5回でも十分。(処理時間を半分にできる)。
- メモリー容量が大きければ、並列ジョブで走らせられる。
 - そのため、8つコアを使用するためには、512 GBのメモリが必要。
- 行列演算の最適化?
- GPU?

まとめ

- 木曾超広視野カメラ(Tomo-e Gozen)のデータ処理に対し、低ランク行列分解の方法を提案した。
- Low rank matrix + Sparse matrix に分解する。
- 実際、Robust PCA や GoDecを用いると、「突発事象」と「背景」を綺麗に分離することができた。
- 現実的な時間で処理できる。
- 重要な情報を落とさずに、データ圧縮できる(10分の1程度: 一晩で30 TB → 3 TB)。
- Tomo-e Prototype Model のデータでテストした。
 - もう少し高速化できるかも(ハードウェアによる)
- 動画天文学に必要な不可欠なツールになるだろう。
 - Hard diskなどの記録媒体に革新的な性能向上がない限り。