2008 年度 芝浦工業大学 システム工学部 電子情報システム学科

総合研究論文

X線天文衛星「すざく」による観測データの デジタルプラネタリウムへの効果的投影法の研究

Making of digital planetarium contents of data by SUZAKU

P05052

さいとう だい **齊藤 大**

指導教員: 久保田あや講師

目次

第1章	はじめに	1
1.1	目的と背景	1
1.2	デジタルプラネタリウム	1
第2章	X 線天文学とすざく衛星	3
2.1	X 線天文学	S
2.2	すざく衛星の概要	3
2.3	搭載検出器	4
	2.3.1 XIS	4
	2.3.2 HXD	5
	2.3.3 性能	
第3章	FITS & WCS	7
3.1	FITS について	7
	3.1.1 FITS の構造	7
	3.1.2 FITS ブラウザ	8
	3.1.3 CFITSIO によるデータの入出力	G
3.2	WCS について	10
	3.2.1 基本 FITS での変換	10
	3.2.2 WCS での基本的な変換手順	11
	3.2.3 投影法の種類	12
	3.2.4 WCSLIB	17
第4章	全天画像の開発	19
4.1	使用する XIS 画像データ	19
4.2	デジタルプラネタリウムへの投影	21
4.3	データの入手	21
4.4	リストファイルの作成	22
4.5	昨年度プログラムの変更点	22
	4.5.1 プログラム実行コマンド	22
4.6	プログラムの実行と結果	23
	4.6.1 プログラムの実行	23
	4.6.2 画像の位置の正しさの確認	23
4.7	実際の投影と問題点・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	25
第5章	効果的上映法の検討と開発	27
5.1	効果的上映法の検討・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	27
5.2	拡大画像作成プログラムの開発	28

	5.2.1 必要な入力パラメータ	29
	5.2.2 ヘッダーキーワードの読み込み	29
	5.2.3 リスト化	30
5.3	結果	31
5.4	photoshop を用いた画像処理	33
5.5	さまざまな天体の拡大画像	35
5.6	プラネタリウム上映番組の提案	37
第6章	まとめと今後の課題	39
付録A	開発プログラム	41
A.1	拡大画像作成プログラム:convertWCS2.c	41
A.2	STIFF	46

第1章 はじめに

1.1 目的と背景

当研究室ではISAS(宇宙科学研究本部)と共同で、X線天文衛星「すざく」による観測データのプラネタリウム投影に向けた研究を進めている。昨年度では西尾光史がX線天文衛星「ROSAT」の全天観測データでX線プラネタリウムの基礎開発を行った。[1]この成果をうけ、本研究では日本の最新のX線天文衛星である「すざく」で得られたデータを元にプラネタリウム投影に向けた研究を行う。「すざく」の観測データは観測提案者にデータが提供された後、1年たつと全世界に公開され、誰でも利用することができる。このデータを簡単に一般の人が観賞できるプラネタリウムで投影することができれば利用価値が増え、また最先端の研究現場と一般の人々をつなぐことができるようになる。これを目的として研究を行っている。

1.2 デジタルプラネタリウム

今までのプラネタリウムでは、光学式プラネタリウムのように原板を用意して表現する方法が 主流である。だがデジタルプラネタリウムでは複数のプロジェクターを使ってデータを投影する ことができる。光学式のように原板を用意するよりも、衛星画像を用いたり、コンピュータグラ フィックスなど使って自由に映像をできる点などプラネタリウム作成がしやすくなっている。ま た、単純に作成がしやすくなったというメリットだけではなく、投影できるものの幅が格段に増 え、プラネタリウム番組の表現力が大幅にアップするといったメリットがある。

図 1.1^1 は投影イメージである。

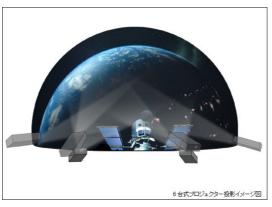


図 1.1: 投影イメージ

¹http://www.ssc.slp.or.jp/planet/planetarium/phenomenon/projector.htm

第2章 X線天文学とすざく衛星

2.1 X 線天文学

ブラックホールや中性子星、銀河団などの強い重力源でガスが捉えられると、ガスは数千万度の高温に熱せられ、X線が放射される。すなわち、宇宙からのX線を観測することで宇宙の高温かつ激しい活動の様子を知ることができる。X線天文学は、観測天文学の一分野で、そのような天体から放射される X 線の研究を行なう。X 線放射は地球の大気によって吸収されるため、X 線の観測装置は高い高度へ位置しなければならなく、X 線検出器は現在人工衛星に搭載されるのが普通である。

2.2 すざく衛星の概要

「すざく」(Astro-E2)[3] は、日本で5番目のX線天文衛星として、2005年7月10日に打ち上げられた。X線源となる高温のプラズマの観測、遠方の銀河団を観測し宇宙の進化についてのデータの提供、ブラックホール候補天体や活動銀河核の観測が目的である。2000年2月、打ち上げ口ケットの不具合によって軌道投入できなかった Astro-E 衛星の再挑戦をかけたミッションである。図 2.1 は「すざく」の外観である。「すざく」は直径 2.1 m、全長 6.5 m(軌道上で鏡筒伸展後)の大きさを持ち、太陽パドルを広げると 5.4 m の幅になる。衛星の重量は 1700 kg にもなり、日本の科学衛星としては、これまでにない大型衛星となった。「すざく」の特性を表 2.1 に表す。さまざまな X 線天体に対して、X 線 CCD カメラ (XIS: 0.3-12keV)と硬 X 線検出器 (HXD: 10-600keV)でこれまでになかった広いエネルギー領域にわたり、より高いエネルギー分解能かつ高感度での観測ができることが最大の特徴である。[4]

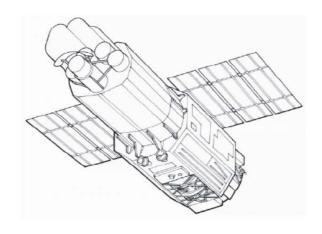


図 2.1: すざく外観

内之浦宇宙空間観測所				
4m)				
$570 \mathrm{km}$				

表 2.1: すざくの特性 (すざくファーストステップガイドより)

2.3 搭載検出器

2.3.1 XIS

X I S は 4 つの C C D の X 線撮像分光計から成っており、X 線源の画像を撮像するものである。 XIS の 4 台のセンサー (図 2.2) をそれぞれ X0, X1, X2, X3 と呼ぶ。また、CCD には表面照射型 (Frontside Illuminated; FI) と裏面照射型 (Backside Illuminated; BI) がある。表面照射型 CCD では X 線を電極側から入射するため、低エネルギーの X 線は電極や絶縁層で吸収されてしまうのに対し、裏面照射型 CCD では X 線を電極の逆側から入射するため低エネルギーの X 線に対して高い検出効率を得ることができる。0.2 keV から 12 keV の X 線領域で、広い視野での撮像を行いながら精度の高い分光を連続的に行うことが可能である。得られる画像視野 $17'.8 \times 17'.8$ に対し、1024 ピクセル $\times 1024$ ピクセルの解像度を持っている。天体の明るさに応じて一回の観測に $1 \sim 3$ 日程度を要する。[3][5]



図 2.2: XIS の外観

2.3.2 HXD

X線望遠鏡でカバーされる X線の何十倍ものエネルギーを持つ硬 X線からガンマ線の領域を観測するため、硬 X線検出器が搭載されている。このように高いエネルギーまで観測できる装置が衛星に搭載されるのは、日本で初めてである。この検出器はガドリニウム・シリケート結晶を用いた無機シンチレータ (GSO) とシリコン検出器を組み合わせたものである。筒状に伸びた井戸型シンチレーターによって周りからの雑音ガンマ線を低減するなど、様々な工夫により、このエネルギー領域で、これまでにない、高感度の観測が可能である。 (図 2.3)[3][6][7]



図 2.3: HXD の外観

2.3.3 性能

表 2.2 にそれぞれの検出器の性能を表す。

	XIS	HXD	
視野	17'.8x17'.8	$34'x34'(\le 100 \text{keV}), 4'.5x4'.5(\le 100 \text{keV})$	
エネルギー帯域	$0.2\text{-}12\mathrm{keV}$	$10\text{-}600\mathrm{keV}$	
エネルギー分解能	~130eV	3.5 keV (10 49 keV)	
———— 時間分解能	8s,7.8ms	$61\mu s$	
センサー当たり有効面積	$1300cm^{2}$	$160cm^2(>30keV), 330cm^2(>40keV)$	
 1 画素のサイズ	$24\mu\mathrm{mx}24\mathrm{m}\mu$		
有効画素	1024×1024		

表 2.2: 検出器の性能

第3章 FITSとWCS

3.1 FITS について

FITS (The Flexible Image Transport System)[2] は天文分野で使われるファイルの代表的フォーマット (形式) である。最初は天体画像のデータを交換するために作成されたフォーマットだったが、いくどかの拡張を経て、現在では、(天体スペクトルなどのデータ、X線観測のイベントデータ、天文カタログを収めた表データ)、などの天文分野で使われる一通りのデータを扱える汎用のフォーマットになっている。 IAU(International Astronomical Union = 国際天文連合)の commission5(天文学データを扱う第5委員会)の下のFITS ワーキンググループがフォーマット規約の管理をしており、この意味でも「お墨付き」の天文標準フォーマットと言える。拡張子はfits。

3.1.1 FITS の構造

単純な FITS ファイルの構造 (図 3.1) は ASCII テキストで書かれたヘッダとバイナリの (通常 は多次元の) データ配列からできている。



図 3.1: 単純な FITS ファイル構造

これから研究で使用していく一般的な FITS ファイルにはこの「基本」FITS 要素に加えて、同じデータ格納構造 (へッダ + データ) を持つ拡張された他の FITS 要素が付け加わってもよいことになっている。模式図で表すと図 3.2 のようになる。



図 3.2: 一般的な FITS ファイル構造

このように $(\land \lor \not = + \vec = - \not = -)$ の FITS 要素が連なっており、いずれの FITS 要素も整数個の (論理) レコードからなる。論理レコードのサイズは 2880 バイト である。

前述のとおり、ひとつの要素は大きくヘッダー部とデータ部の2つ部分に分けられる。

前半部のヘッダはヘッダレコードとも呼ばれ、データの目的、種類、構造、バイト数、レコード数などのデータに関する解説部分となっている。 1 行が 80 文字 1 からなるカードイメージの連なりで、整数個の論理レコードに収められる (1 レコードには 2880/80=36 行が収まるので、ヘッダが 36 行を越える時は複数個の論理レコードが必要となる。また、ヘッダが 1 レコードに満たない場合は空白行で埋められちょうど整数個のレコードとされる。) 1 つの行の各欄の使い方や用語には一定の規約がある。

次の図 3.3 は FITS ファイルを閲覧した画面である。[2] ヘッダ部とテーブル部のペアが、複数の extension として連なっているのがわかる。この header+table のペアは、一番上の欄の「extension」を展開したものである。

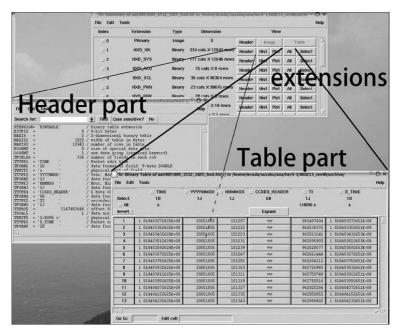


図 3.3: FITS ファイルの閲覧

3.1.2 FITS ブラウザ

FITS ファイルブラウザとして、ds9[?][1] を使用する。ds9 は SAO (Harvard 大の付属天文台である Smithsonian Astronomical Observatory) で開発されている FITS ブラウザである。 代表的な機能は次のとおりである。

- FITS ヘッダの閲覧
- Binary Table, Image などの Extension を扱える
- FITS の n 次元画像を扱える
- 圧縮された FITS 画像を扱える
- 生のイベントファイルをサポート
- IRAF 形式をはじめ一般ファイルの閲覧と編集も可能
- ブリンク、コントア、スライス面のプロットなどができる

- ビニングやグリッド表示ができる
- WCS に対応 (天球座標表示可能)
- DSS サーバに接続して表示可能

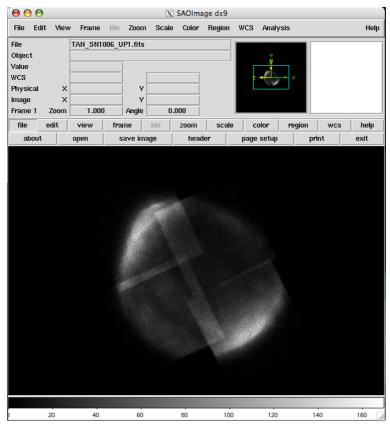


図 3.4: ds9:使用例

3.1.3 CFITSIOによるデータの入出力

cfitsio[1] は FITS ファイルの読み書きのためのサブルーチンインターフェイスである。各ルーチンは C と FORTRAN で書かれ各種コンピュータで実行可能である。FITS の最新の規約に対応するようにアップデートされており、standard extension、多次元配列、可変長配列の Binary Table、WCS や checksum にも対応している。 UNIX/VMS と Windows に対応するソースコードと、Windows の DLL (Visual C++, Borland C++ 用) も配布されている。関連して FITS ファイルのベリファイツールも公開されている。

最新バージョンは2009年1月現在で3.100である。情報、ソフトウェアは下記から入手できる。

http://heasarc.gsfc.nasa.gov/fitsio/

3.2 WCS について

現行の FITS スタンダード (NOST 100-2.0) では座標表現に関しては簡単な変換に対応したいくつかのキーワード (CRVALn, CRPIXn, CDELTn, CTYPEn, CROTAn) しか定義されておらず、実際の天球座標とデータ配列の間の対応を表現するには不十分な点があった。それを補うためにより一般的な表現方法として提案されたのが WCS (World Coordinates System) [8] である。

ここで World Coordinate (世界座標) とは、多次元のパラメータ空間のうち何らかの物理的測量値、例えばスペクトル中の波長値や物理空間中の方向を表す緯度経度などを提供する座標のことを指し、世界座標と FITS ファイル中の N 次元データ配列の各データ値を対応させるためのキーワードなどを含む規程集が WCS である。

2002 年 ~ 2005 年に World Coordinate の表現方法を扱った Paper (WCS Paper I)[8] 、天球座標の表現を扱った Paper (WCS Paper II)[9]、スペクトル関係の Paper (WCS Paper III)[10] が正式に IAU FWG で認められ、機器関係の歪みを扱った Paper (WCS Paper IV)[11] もドラフト段階のものが公開されている。

3.2.1 基本 FITS での変換

当初の FITS(基本 FITS) では、データ配列のインデックス (i,j,k,\cdots) から物理量である座標値 へ (x_i,x_j,x_k,\cdots) の変換のために以下のキーワードが定義されている。 [2]

CRVALn 参照点での座標値

 CRPIXn
 参照点でのインデックス

 CDELTn
 参照点での座標値の増分

 CTYPEn
 座標軸の種類 (8 文字)

CROTAn 回転角

(n は座標軸の番号、単位は、SI 系と角度の「度」)

これにより、CROTAn=0.0 の場合、座標値 x_n はインデックス n から次式で計算される。

$$x_n = CRVALn + CDELTn \times (n - CRPIXn) \tag{3.1}$$

これはあまりにも単純であり、もっと一般的な表現方法として WCS が提案された。

3.2.2 WCS での基本的な変換手順

WCS の提案 (WCS Paper I)[8][2] では、ピクセル座標から WCS への変換は、複数のステップ (3 つの変換) を踏んで変換されることになる。ピクセル座標 (p_j) 中間ピクセル座標 (q_i) 中間 世界座標 (x_i) 世界座標、である。このステップの流れ図と簡単な説明は次のようになる。

ピクセル座標

(step1) 線形変換する (CRPIXjs, PCi_js or CDi_js キーワード) 行列を掛け回転、歪み、(オプションで) スケールの補正

中間ピクセル座標

(step2) 物理単位へ再スケーリングする (CDELTis キーワード)

中間世界座標

(step3) 座標変換 (CTYPEis, CRVALis, PVi_ms キーワード) 球面から平面への射影と、実世界座標への変換

世界座標 (World Coordinate)

• 最初のステップ (step 1) は、ピクセル座標から中間ピクセル座標への線形変換である。この ためにはピクセル座標ベクトル p_i に対して行列を掛ける。

$$q_i = \sum_{j=1}^{n} m_{ij} (p_j - r_j)$$
(3.2)

ここで r_j は CRPIXj で与えられる参照点でのピクセル座標であり、 m_{ij} が変換行列、 q_i が中間ピクセル座標である。これ以降、添字の j はピクセル軸を、i は世界軸を表す。 m_{ij} は NxN の正方行列であり、N は NAXIS キーワードで与えられる。ただし、この点は WCSAXES キーワードによってより一般化される。変換結果の q_i は、中間世界座標軸と一致する方向の中間ピクセル座標軸ベクトルであり、無次元のピクセル単位での参照点からのオフセットである。

• 従って 2 番目のステップ (step 2) である、 q_i を対応する中間世界座標の x_i に変換するには、単に次のようなスケーリングをするだけである。

$$x_i = s_i q_i \tag{3.3}$$

• 3番目のステップ (step3) は中間世界座標から WCS への変換である。具体的には、球面から平面への射影法と、平面と天球面の接点での世界座標の値から実際の世界座標 (WCS) に変換する。この変換は CTYPEi に依存する。単純な線形軸では、 は CRVALi で与えられる参照点における座標値に加えるオフセットと解釈される。それ以外の場合には、CTYPEi は x_i , CRVALi と他のパラメータの関数を規約に従って定義することになる。規約にない CTYPEi は線形と解釈される。非線形座標は CTYPEi に "4-3" 形式で記述される。これは 例えば 'ABCD-XYZ' のようなもので、最初の 4 文字が座標の種類を表し、5 番目の文字は '-' で、残りの 3 文字が中間世界座標から世界座標に変換するアルゴリズムを指定する。座標の種類が 4 文字に満たない場合は '-' で補い、アルゴリズムが 3 文字に満たない場合は空白を補う、'RA-UV' のように。ただし、アルゴリズムのコードは 3 文字にすることを推奨する。

3.2.3 投影法の種類

AIT: Hammer-Aitoff 投影法

ランベルト正積方位図法の横軸図を変形して擬円筒図法のような形状にしたものである。角の 歪みが非常に大きくなっている。Aitoff が ARC(Zenithal equidistan 投影法) を元に考え出したも のに、Hammer が修正を加えたものである。この投影法の定義式を以下に示す。

$$x = 2\gamma \cos\theta \sin\frac{\phi}{2} \tag{3.4}$$

$$y = \gamma sin\theta \tag{3.5}$$

$$\gamma = \frac{180^{\circ}}{\pi} \sqrt{\frac{2}{1 + \cos\theta \cos\frac{\phi}{2}}} \tag{3.6}$$

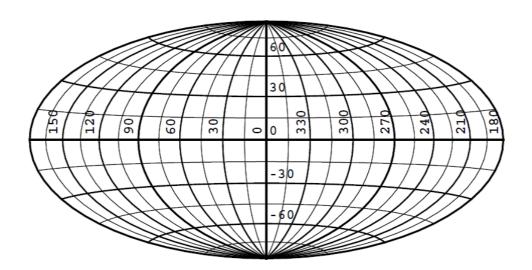


図 3.5: Hammer-Aitoff 投影法

TAN:Gnomonic 投影法

この投影法は心射図法ともよばれ、図 3.6 のように接点を中心に置きすべての大円を直線に投影する図法である。光学天文学では広く使用されており、AIPS の中にある心射図法のコードであることから、TAN とよばれる。次に定義式を示す。

$$R_{\theta} = \frac{180^{\circ}}{\pi} \cot \theta \tag{3.7}$$

$$\theta = tan^{-1} \left(\frac{180^{\circ}}{\pi R_{\theta}}\right) \tag{3.8}$$

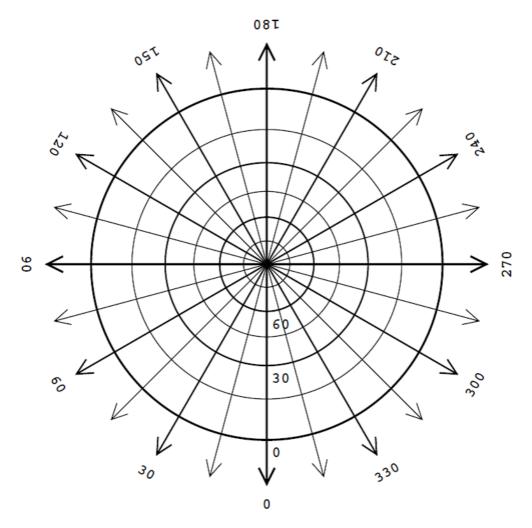


図 3.6: Gnomonic 投影法

TSC:Tangential spherical cube 投影法

この投影法は、図 3.7 のように極方向の図とそれ以外 (赤道方向) とに分けられており、この展開図を組み立てると、ちょうど立方体の内側に全天画像を貼り付けたようになる。この投影法は、プロジェクタが中央に 1 台と、四方に 1 台ずつあるプラネタリウムで使用され、本研究での全天画像作成に使用する。

定義式を以下に示す。

$$x = \phi_c + 45^{\circ} \chi \tag{3.9}$$

$$y = \theta_c + 45^{\circ} \psi \tag{3.10}$$

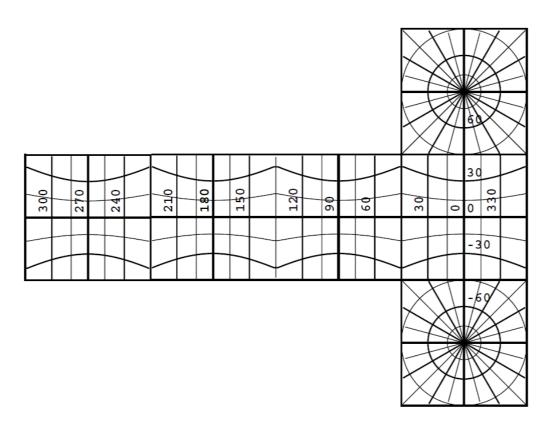


図 3.7: Tangential spherical cube 投影法

ZEA:Zenithal equal-area 投影法

ZEA とはランベルト正積方位図法とも呼ばれる地図投影法の一つ。極を基準点 (中心) とした場合、経線は中心から放射状に、緯線は基準点を中心とする同心円上に描かれる (図 3.8)。緯線の間隔は特に図の外側で狭くなっている。中心付近の歪みは比較的小さい。今回は、横軸法 (赤道方向から見た図) を用いて画像を作成した。この投影法の式を以下に示す。

$$R_{\theta} = \frac{180^{\circ}}{\pi} \sqrt{2(1 - \sin\theta)} \tag{3.11}$$

$$=\frac{360^{\circ}}{\pi}(\frac{90^{\circ}-\theta}{2})\tag{3.12}$$

$$\theta = 90^{\circ} - 2\sin^{-1}(\frac{\pi R_{\theta}}{360^{\circ}}) \tag{3.13}$$

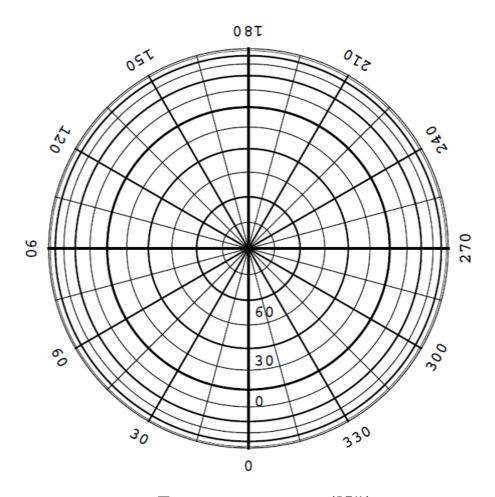


図 3.8: Zenithal equal-area 投影法

CAR:Plate Carree 投影法

経緯線が直角、等間隔である正距円筒図法のひとつで、標準緯度を 0 度に置いたものである。この図法は緯度、経度をそれぞれ図の縦と横にそのまま読み替えたもので、標準緯線上と縦方向に関しては正距である。標準緯線から離れるほど横方向の長さが拡大されるため、角度は正しくない。定義式を以下に示す。この投影法のイメージは、図 3.9 である。

$$x = \phi(\mathbf{\textit{\textbf{経}}}\mathbf{\textit{\textbf{E}}}) \tag{3.14}$$

$$y = \theta(緯度) \tag{3.15}$$

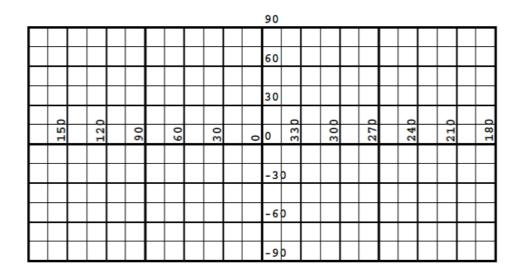


図 3.9: Plate Carree 投影法

PCO:Polyconic 投影法

正規多円錐図法とも呼ばれ、擬円錐図法の一種である。緯線は中央経線の延長線上に中心を持つ異心円群として表される。経線は緯線を等間隔に区切った点を結んだ曲線になる。緯線の長さは正しくなるが、経線の長さは中央経線から離れるにつれて引き延ばされていく。各緯線上と中央経線上の距離は正しくなっている。

この投影法の定義式を以下に示す。

$$x = \frac{180^{\circ}}{\pi} \cot\theta \sin(\phi \sin\theta) \tag{3.16}$$

$$y = \theta = +\frac{180^{\circ}}{\pi} \cot\theta [1 - \cos(\phi \sin\theta)]$$
 (3.17)

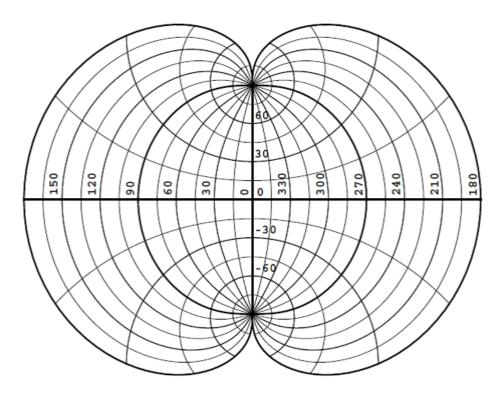


図 3.10: Polyconic 投影法

3.2.4 WCSLIB

WCSLIB は ANTF(Australia Telescope National Facility) の M. Calabretta によりリリース された WCS を扱うライブラリで、WCS の Paper II で提案されている座標変換をインプリメントしたルーチン集である。WCSLIB のソースコードは ANTF で公開されており、以下のサイトから入手することが出来る。(2009 年 1 月時点で WCSLIB 4.3 が最新版)

ftp://ftp.atnf.csiro.au/pub/software/wcslib/wcslib.tar.gz

また、WCSLIB の詳細については以下のサイトを参照すると良い。

http://www.atnf.csiro.au/people/mcalabre/WCS/

第4章 全天画像の開発

4.1 使用する XIS 画像データ

画像がとれる XIS 検出器で得られたデータを使用する。観測データは、JAXA / 宇宙情報解析 研究系で運用している JUDO と呼ばれるシステム用に保存されているものを使用。もとのデータ は XIS0,1,2,3 それぞれについて 0.2-12keV に 4096 チャンネル割り当てられている。JUDO 用の画像はこれを 0.4-1.5keV,1.5-4keV,4-10keV についての天空座標での画像を作成し、これを 4 台の XIS で足し合わせている。[3][4]

使用データの特徴の表と、1ファイルの例として、天体 SN1006 の一部が次である。

画像サイズ	256x256 ピクセル
視野	17.8x17.8 分角
形式	img で圧縮保存
投影タイプ	TAN

表 4.1: 得たファイルの特徴

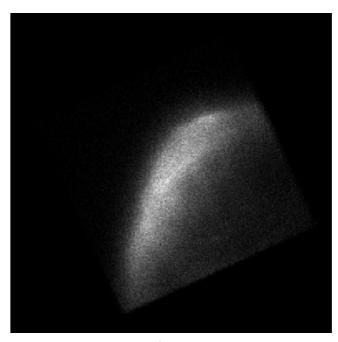


図 4.1: soft バンド [0.4-1.5keV] ファイル名:suzaku_ 100019010_ soft.img

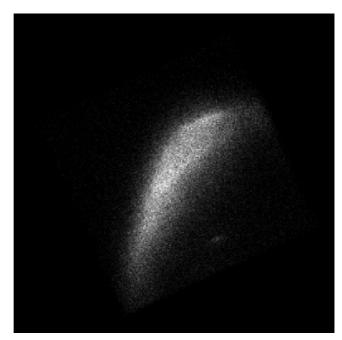


図 4.2: medi バンド [1.5-4keV] ファイル名:suzaku_ 100019010_ medi.img

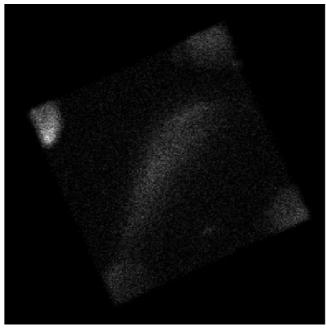


図 4.3: hard バンド [4-10keV] ファイル名:suzaku_ 100019010_ hard.img

4.2 デジタルプラネタリウムへの投影

6台のプロジェクターで投影する場合の投影法を考える。投影法には全天を六枚に分けた TSCが適しており、使用する。デジタルプラネタリウム 1 では、一台で投影する画像 (図 4.4) の解像度の限界が 2048×2048 ピクセルとなっている。

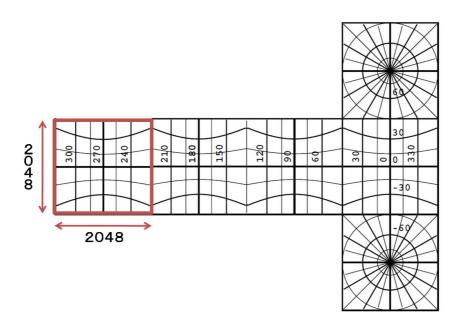


図 4.4: 投影法 TSC での一台の画像分

4.3 データの入手

使用するデータは

ftp://ftp.darts.isas.jaxa.jp/pub/suzaku/qlproducts/judo_images

でダウンロードをする。 ダウンロードには curl コマンドを使用

% curl --proxy proxy.se.shibaura-it.ac.jp:10080 -0 ftp://ftp.darts.isas.jaxa.jp/pub/suzaku/qlproducts/judo_images.tar

当研究室で行う場合

公開データが 0.4-1.5keV、1.5-4 keV、4-10keV の 3 バンドに分けられて置いてあり、全世界に公開されている。現在のファイル数は 784 である。

¹実際の投影を行った科学技術館での場合

4.4 リストファイルの作成

次に、ダウンロードしてきたファイルのフルパス付きのリストファイルが必要となる。soft,medi,hard ごとにファイル分けをし、次のコマンドを入力するとリストファイルが作成される。(soft の場合)

% find (soft/までのフルパス名) -print > soft.list

この手順で soft, medi, hard ごとにリストファイルを作成する。中身は次のようになる。

 $/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100001010_soft.img\\/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100001020_soft.img\\/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100002010_soft.img$

•

.

なお、リストファイルはプログラムと同じディレクトリに置く。

4.5 昨年度プログラムの変更点

公開データを全天に貼り付けるために、ROSAT 全天サーベイ用に開発されたプログラム $^2[1]$ をもとににすざく用にパラメータを変更して使用した。変更する点は解像度の変更である。ROSAT ではひとつ 512x512 ピクセルであったものを、「すざく」では 256x256 ピクセルであったため、プログラム内にある 512 ピクセルのキーワードにあたる部分を $512 \rightarrow 256$ へ変更していった。

4.5.1 プログラム実行コマンド

プログラムを実行する際のコマンドは以下のようになっている。[1]

%./convertWCS (list ファイル名) PRJ outimagesizeX outimagesizeY crpix1 crpix2 pixelsize (保存先 fits ファイル名)

PRJ 投影法

outimagesize出力ファイルの解像度crpix参照ピクセル座標

pixelsize 座標スケール (degree/pix)

²http://plain.isas.jaxa.jp/ ebisawa/Planetarium/convertWCS.c

4.6 プログラムの実行と結果

4.6.1 プログラムの実行

まずはじめに、変更したプログラムを使って、投影法 TSC で soft バンドのデータを用いて 2048x2048x6 のパターンをプログラム実行、全天図を作成した。プログラム実行コマンドは以下である。

%./convertWCS soft.list TSC 8192 6144 7168.5 3072.5 0.0439453125 (保存先 fits ファイル名)

中心位置はそれぞれ (7168,3072)/pix である。図 4.5 では見にくいため、緑のコントアを表示させている。

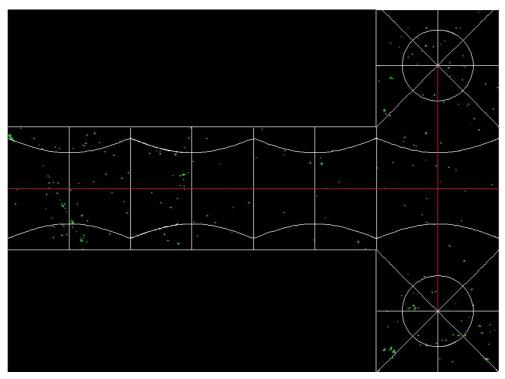


図 4.5: 得られた全天画像

4.6.2 画像の位置の正しさの確認

変更したプログラムで作ったデータが全天の正しい位置に置かれているか確認をした。手順としてまず 1 データを確認するように実行した。1 データを全天に置いて部分的に拡大したものと、その 1 データ単体をそれぞれ ds9 で表示させた。ds9 ではカーソル部分の座標が表示でき、ふたつのデータのそれぞれ四隅に (図 4.6) マウスでカーソルをあわせて、位置を照らし合わせていった。 照らし合わせた結果、正しい位置に置かれていることが確認できた。

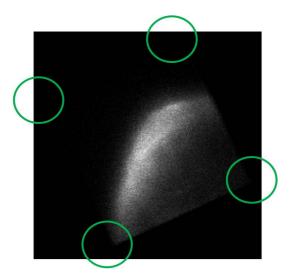


図 4.6: 四隅の座標確認

照らし合わせた結果、正しい位置に置かれていることが確認できたため、キーワードを変更したプログラムは「すざく」の場合でも正常に動作することが証明できた。次はできた全天画像のヘッダーである。

```
SIMPLE =
                          T / file does conform to FITS standard
BITPIX =
                        -32 / number of bits per data pixel
NAXIS
                          2 / number of data axes
NAXIS1 =
                        8192 / length of data axis 1
                        6144 / length of data axis 2
NAXIS2 =
EXTEND =
                          T / FITS dataset may contain extensions
COMMENT FITS (Flexible Image Transport System) format is defined in 'Astronomy
         and Astrophysics', volume 376, page 359; bibcode: 2001A&A...376..359H
COMMENT
RADESYS = 'FK5
                            / Coordinate system
EQUINOX =
                   2.000E+03 / Equinox
CTYPE1 = 'RA---TSC'
                            / DEC projection
CTYPE2 = 'DEC--TSC'
                            / DEC projection
CRPIX1 = 7.1685000000000E+03 / X reference pixel
CRPIX2 = 3.0725000000000E+03 / Y reference pixel
CRVAL1 = 0.000000000000000E+00 / RA of reference pixel
CDELT1 = -4.3945312500000E-02 / X pixel increment (degree)
CDELT2 = 4.3945312500000E-02 / Y pixel increment (degree)
END
```

キーワード説明

SIMPLE 論理値 T又は Fを持ち、この規格に適合するかどうかを示す。

BITPIX 整数値を持ち、データのビット数を表わす。

NAXIS 999 以下の非負の整数値を持ちデータ配列中の軸の数を表わす。

NAXISn 非負の整数値を持ち、n番目の軸の大きさを表わす。

EQUINOX ヘッダまたはデータ中に与えられた位置を表わすのに使われた座標系に対する equinox

を年単位で浮動小数点表示したもの。

CTYPEn 局所球面座標各軸のタイプ。投影法の種類がわかる。

CRPIXn参照ピクセル座標。投影法における中心のピクセル座標。

CRVALn CRPIXn で示される参照点が、CTYPEn で示される座標上のどの値に対応するかを表

わす浮動小数点値を持つ。

CDELTn CRPIXn 参照点で評価された、ピクセルインデックスに関する CTYPEn で表わされる

座標の増分を与える浮動小数点値。

END 値を持たず 9-80 桁は空白。ヘッダの終わりを表わす。

4.7 実際の投影と問題点

2008 年 8 月 29 日、実際に科学技術館において作成した全天画像の投影を行った。

「あかり」や ROSAT の場合と異なり、すざくは全天サーベイを目的とする衛星ではないため $(2.2\ \mbox{ 節、}3\ \mbox{ ページ})$ 、図 $4.5\ \mbox{ では }X$ 線画像が空全体に広がっているような X 線の星空を眺められる というわけにはいかず、空にすざくの <u>視野</u> である $17.8x17.8\ \mbox{ 分角 }($ 月の直径の約半分) の四角 が散らばっているだけにしか見えなかった。これでは、プラネタリウムに上映するコンテンツと してあまりにも迫力がなさすぎた。

第5章 効果的上映法の検討と開発

5.1 効果的上映法の検討

4章で示したように、全天観測衛星ではない「すざく」の画像をプラネタリウムで上映するには、単に全天にデータをちりばめただけではプラネタリウムの上映が成り立たないことがわかった。しかし、「すざく」衛星は日本の誇る最新の X 線天文衛星であり、プラネタリウムを通してその成果の一端を知ってもらうことは、最新の宇宙像への興味をかき立てる糸口となることは間違いない。

では、どうすれば「すざく」の成果をわくわくしながら理解できるような宇宙番組がプラネタリウムで提供できるだろうか? 2章で述べたように、「すざく」は全天観測を行うのではなく、指向観測を行っている。これは、興味深い天体を集中して高い感度で観測し、その天体のメカニズムを明らかにする観測である。つまり、すざくの成果を知るということは、興味深い天体をすざくとともに「じっくり観る」というところに最大の醍醐味がある。これをプラネタリウムで体感するには、やはりある天体に着目し、プラネタリウム上で拡大して表示させてはどうかと考えた。

しかし、作成した全天画像をただ拡大させては図 5.1 のように荒くなってしまう。 4.2 で示したように、全天画像の解像度はこれが限界のため、全天画像とは別のファイルの拡大した部分のみの画像を作成することにした。別に用意して投影するため、改造度に関して考慮する事はなくなった。

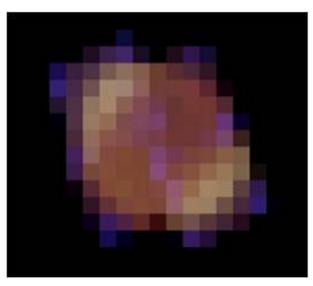
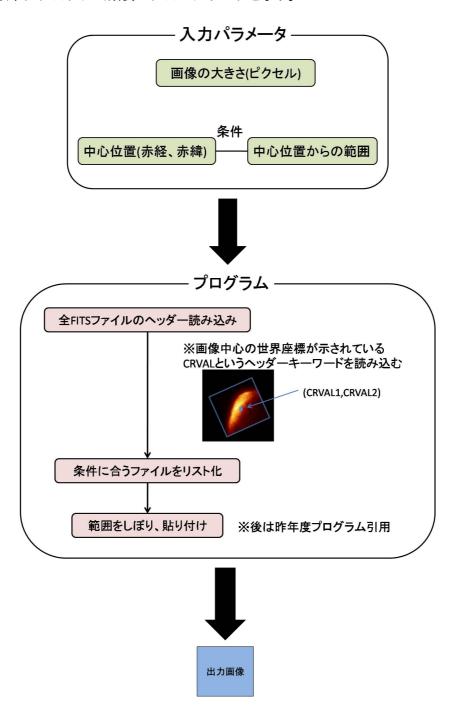


図 5.1: 単純な拡大画像 例:天体 SN1006

5.2 拡大画像作成プログラムの開発

拡大画像を作成するにあたって、全天画像を作成する際に使用したプログラム 1 を改良することになった。以下にプログラム改良のフローチャートを示す。



¹http://plain.isas.jaxa.jp/ ebisawa/Planetarium/convertWCS.c

まずはじめに入力パラメータとして画像のサイズ、さらに条件として中心位置 (赤経、赤緯)、中心位置からの範囲を入力。それを受け取ったプログラムはリストファイルから全 FITS ファイルを開いた後、全 FITS ファイルのヘッダーを読み込んでいく。ヘッダーの読み込みの際に、画像中心の世界座標が示されている [CRVAL1]、[CRVAL2] キーワードの値を扱えるよう読み込みを行う。後は先ほどの条件に合うファイルをリスト化し、範囲をしぼり、貼り付け 2 を行う。ここで貼付ける作業は昨年度のプログラムを利用することにした。また、投影法は JUDO の元画像に習い、TAN 投影法とした。

5.2.1 必要な入力パラメータ

プログラム実行時に必要なパラメータを次のように設定した。

%./convertWCS2 (list ファイル名) PRJ outimagesizeX outimagesizeY crval1 crval2 range1 range2 pixelsize (保存先 fits ファイル名)

座標単位は degree である。

range1,range2 はそれぞれ crval1,crval2 から ±x.xx の範囲ということを示している。

例として天体 SN1006[中心位置 (225.72723, -41.9084) という超新星残骸について、中心から赤経 $\pm 2^\circ$ 、赤緯 $\pm 1^\circ$ の拡大画像を作成する過程を示す。また、投影法は JUDO の元画像に習い、TAN 投影法とした。

%./convertWCS2 soft.list TAN 1360 1360 225.72723 -41.90840 2 1 0.001737 soft.fits

5.2.2 ヘッダーキーワードの読み込み

中心座標は、出力ファイルのヘッダーに入力パラメータを書き込むことで指定した。座標については、接点の天球緯度、経度、つまり天球座標においてのこの画像データの位置を示す [CR-VAL1][CRVAL2] キーワードを全 FITS ファイルのヘッダーから読み込む。読み込む際には、CFIT-SIO の関数でその値を扱えるように読み込む。下記は、入力パラメータの中心座標を

とした出力画像ヘッダーの一部 (例:天体 SN1006) である。

CRVAL1 = 2.2572723000000E+02 / RA of reference pixel<math>CRVAL2 = -4.1908400000000E+01 / DEC of reference pixel

²リスト化した後の貼り付け作業は、昨年度プログラムを引用する。

5.2.3 リスト化

SN1006 の中心座標から、もうひとつの条件の範囲指定である range1 = 2, range2 = 1 に合うファイルのリストを作成し、昨年度プログラムを利用して拡大画像を出力する。

SN1006 の場合、

```
223.72723 < CRVAL1 < 227.72723
-42.90840 < CRVAL2 < -40.90840
```

の範囲条件に合うファイルをリスト化することになる。 実際のファイルには、

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019010_soft.img

Keyword CRVAL1:226.704

Keyword CRVAL2:-41.3998

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019020_soft.img

Keyword CRVAL1:225.964

Keyword CRVAL2:-41.7797

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019030_soft.img

Keyword CRVAL1:225.505

Keyword CRVAL2:-42.0698

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019040_soft.img

Keyword CRVAL1:224.655

Keyword CRVAL2:-42.4005

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019050_soft.img

Keyword CRVAL1:225.496

Keyword CRVAL2:-42.0706

/Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_100019060_soft.img

Keyword CRVAL1:224.647

Keyword CRVAL2:-42.4025

 $/ {\tt Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_500016010_soft.img}$

Keyword CRVAL1:225.866

Keyword CRVAL2:-42.0517

 $/ Groups/workgroup/planetarium/data/SUZAKU/soft/suzaku_500017010_soft.img$

Keyword CRVAL1:225.636

Keyword CRVAL2:-41.8003

この8ファイルがリスト化される。

5.3 結果

先程のパラメータ (天体 SN1006) で実行した場合、図 5.2 のような画像が得られた。解像度にも問題なく画像が貼り付けられているのがわかる。しかし、この図はでは貼り付けたすべてが見にくいため、 \log で表した。(図 5.3)

図 5.3 のとおり、リスト化されたファイルが画像に貼り付けられていることがわかり、拡大画像の作成に成功した。

さらに、同様に medi バンド、hard バンドごとの拡大した FITS ファイルを作成する。できた同じ範囲の3つの FITS について、FITS ブラウザ ds9 によるコマンド

% ds9 - rgb

で三色合成 (疑似カラー合成) を行う。実際の $\mathrm{ds}9$ での三色合成を行うコマンドは以下のとおりである。

% ds9 -rgb -red soft.fits -green medi.fits -blue hard.fits

結果、図5.4の画像が得られた。

なお、図 5.5 は hard バンドを除いた場合である。SN1006 の場合、hard バンドに見られるのはほぼノイズばかりのため、プラネタリウムとして見やすくするために除外をした。

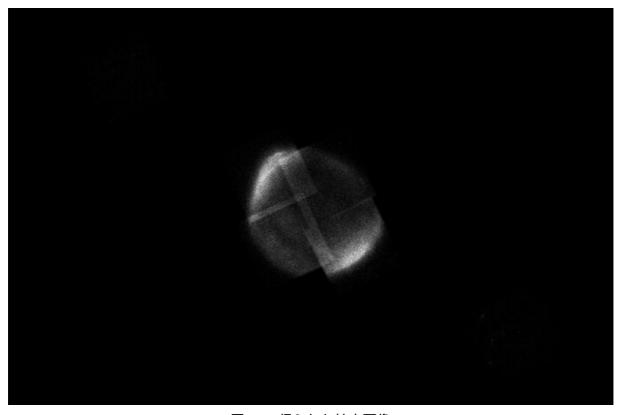


図 5.2: 得られた拡大画像 例:天体 SN1006...soft バンド

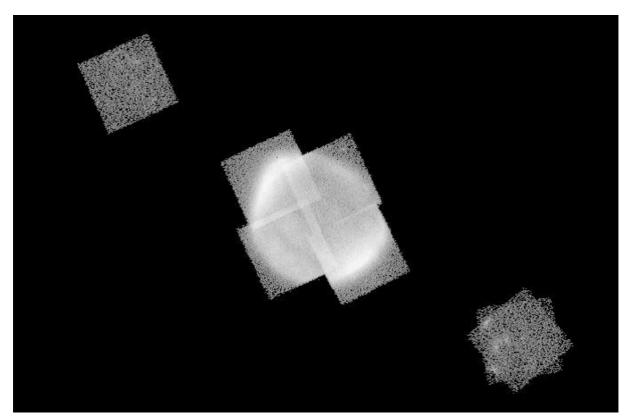


図 5.3: log で表記

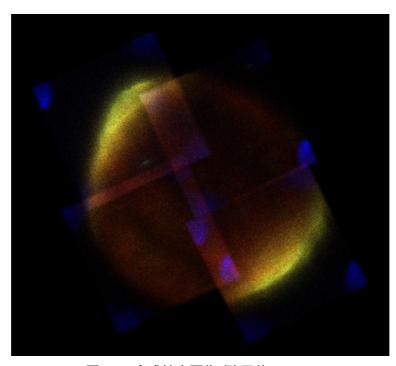


図 5.4: 合成拡大画像 例:天体 $\mathrm{SN}1006$

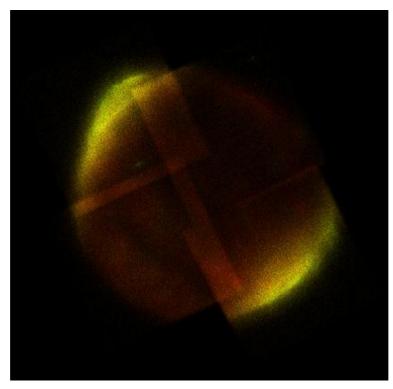


図 5.5: soft バンド+medi バンド

5.4 photoshopを用いた画像処理

この天体の場合、4 つの観測が合成されており、その重なりの部分が目立つ。この重なりが生じている部分に、画像編集ソフト $photoshop^3$ による部分的なレベル補正を図 5.6 のように手動範囲選択して行った。

図 5.7 はレベル補正を行った後の画像である。手動での除去であるため、完全とは言えないが補 正前と比べて天体自体の構造がよくわかるようになった。

³http://www.adobe.com/jp/joc/pepe/diy/

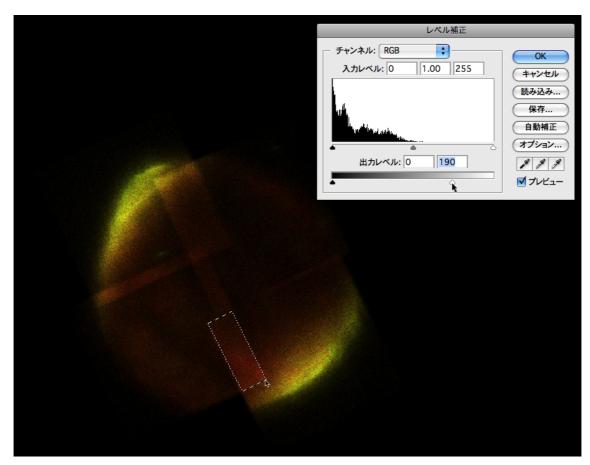


図 5.6: 範囲選択

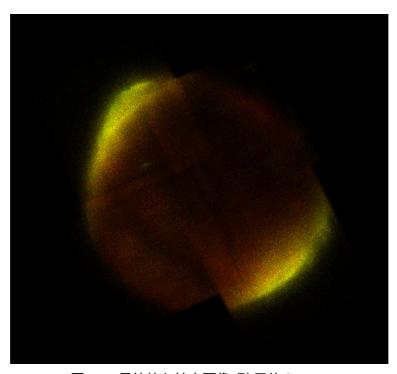


図 5.7: 最終的な拡大画像 例:天体 SN1006

5.5 さまざまな天体の拡大画像

 ${
m SN}1006$ 以外にも様々な天体の拡大画像を作成した。 ${
m SN}1006$ の場合とはことなり、図 5.11 などは天体自体が硬 ${
m X}$ 線(ハードバンド)で明るいため、全体として青くなっている様子が分かる。

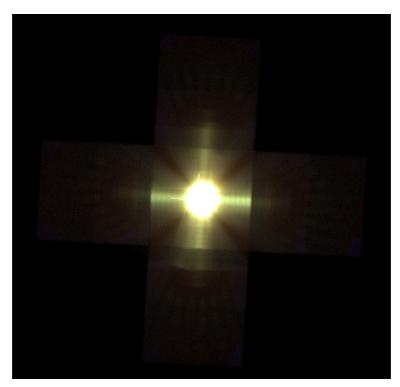


図 5.8: かに星雲 中心座標 (83.66134,21.97525)

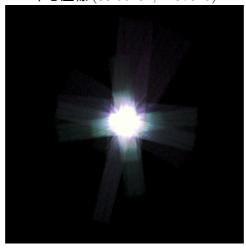


図 5.9: はくちょう座 X-1 中心座標 (299.5907,35.1892)

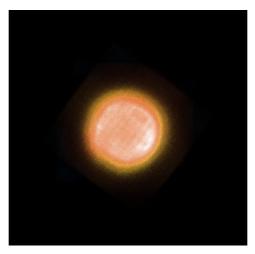


図 5.10: SN 1572 中心座標 (6.3244,64.1507)

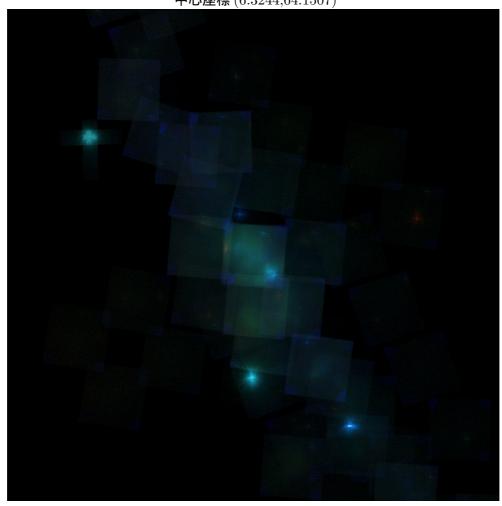


図 5.11: **いて座** 中心座標 (266.5146,-28.9267)

5.6 プラネタリウム上映番組の提案

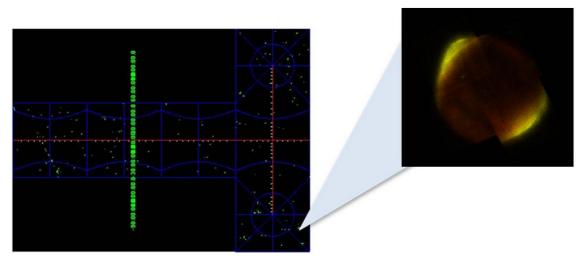


図 5.12: プラネタリウム番組

最終的にプラネタリウムを上映する際には、ナレーションが同時に必要となってくる。流れとしてはまず全天画像の投影。そこから位置をマークし、ナレーションと同時に拡大画像を投影することで、本格的なプラネタリウムを楽しむことができると考えた。

図 5.12 のように投影を行いながら、

「西暦 1006 年に、史上最も明るい星として超新星 $\mathrm{SN}1006$ が輝きました。このおおかみ座領域に位置し、地球からおよそ 7000 光年の距離に存在しています。この画像はその超新星の残骸の約 1000 年後の姿です。

この爆発が実際に起きたのは、西暦 1006 年に地球に光が到達した 7000 年前、つまり今から 8000 年以上も前ということになります。残骸の中にある衝撃波は粒子を極限まで加速させ、宇宙 空間を飛び交う高エネルギーの放射線を発生させていると考えられています。.....」

といったナレーションを行える。

このナレーションは

http://antwrp.gsfc.nasa.gov/apod/ap080704.html を参考として作成した。

第6章 まとめと今後の課題

昨年度西尾光史による総合研究論文「ROSAT 衛星によるX線プラネタリウムの基礎開発」の成果をうけ、「すざく」の全天画像を作成、投影することが目的であった。しかし、投影による結果、プラネタリウムとしては効果的ではなく、新たな投影手段が必要となった。改善策の検討を行い、拡大画像の作成が次の目的となった。結果、おもしろい天体に注目して拡大画像を作成するプログラムの作成、見やすさの改善ができた。

全天画像では迫力に欠けていたが、拡大画像の準備をすることによってプラネタリウムも見やすくなった。そしてすざくによるデータはこれからも更新され続けるため、この手法を用いてより多くの天体画像を用意して、よりおもしろいプラネタリウム番組の提供を行うことができるようになった。

しかし目的のであった拡大画像自体はできたが、その画像作成について不完全な部分が多い。まず、photoshop の手動による画像処理はあまり効率的ではない。そして、画像処理を行った画像も天体によってはどうしてもキレイに写せない。原因としては画像処理の問題だけではない。元のデータの位置自体がバラバラで大きな空きがあったり、ほぼ同じ位置のデータが複数あることで拡大画像作成する際に全て貼り付けてしまう。その重なりが多すぎると、どうしても天体の形がわかりずらくなってしまう、などいろいろな課題が残った。

今後はこれらの様々な問題を解決し、より天体の構造がわかる画像作成すること。より多くの 天体画像を用意すること。実際にプラネタリウムに投影して結果を得ること。さらに他の天文衛 星の観測データでも利用することができるようにすること、などが考えられる。

付録A 開発プログラム

A.1 拡大画像作成プログラム:convertWCS2.c

```
#include <stdlib.h>
#include <string.h>
#include </usr/local/xray/cfitsio/fitsio.h>
#include </usr/local/xray/cfitsio/longnam.h>
#include <wcshdr.h>
#include <wcs.h>
#include <math.h>
main(int argc, char **argv){
  FILE *fp;
  char infile[256], inputlist[256], PRJ[4];
  char outfile[256], RAproj[9], DECproj[9];
  long outimagesizeX, outimagesizeY;
  fitsfile *fptr_in, *fptr_out;
  int status=0, ncards, relax, nreject, nwcs, ctrl, i, j, anynul, num;
  long naxes[2];
  float *inimage, *outimage, X_in, Y_in, X_out, Y_out;
  char *header;
  struct wcsprm *wcs_in,*wcs_out;
  double pixcrd_in[1][2], imgcrd[1][2], phi[1], theta[1], world[1][2], alpha, delta;
  double crpix1, crpix2, pixcrd_out[1][2], crval[2], x, y, a, b;
  int i_out, j_out, nfile;
  long imagesize;
  double pixelsize;
  if(argc !=11){
    printf("usage: convertWCS2 inputfilelist PRJ outimagesizeX outimagesizeY crval1
           crval2 range1 range2 pixelsize outfitsname\n");
    printf("where PRJ must be one of the following; CAR, AIT, ZEA, CSC, TSC, PCO,
           TAN.\n");
    exit(1);
  }else
    {
      strcpy(inputlist, argv[1]);
```

```
strcpy(PRJ, argv[2]);
     sscanf(argv[3], "%ld", &outimagesizeX);
     sscanf(argv[4], "%ld", &outimagesizeY);
     sscanf(argv[5], "%lf", &x);
     sscanf(argv[6], "%lf", &y);
     sscanf(argv[7], "%lf", &a);
     sscanf(argv[8], "%lf", &b);
     sscanf(argv[9], "%lf", &pixelsize);
     strcpy(outfile, argv[10]);
   }
 crpix1 = outimagesizeX / 2 + 0.5;
 crpix2 = outimagesizeY / 2 + 0.5;
 printf("input file list = %s\n", inputlist);
 if(strcmp(PRJ, "CAR")!=0 &&strcmp(PRJ, "AIT")!=0&&strcmp(PRJ, "ZEA")!=0&&
    strcmp(PRJ, "CSC")!=0&&strcmp(PRJ, "PCO")!=0&&strcmp(PRJ, "TSC")!=0&&
    strcmp(PRJ, "TAN")!=0){
   printf("PRJ must be one of the following: CAR, AIT, ZEA, CSC, PCO, TSC, TAN\n");
   exit(1);
 }
 /* 入力画像の ASCII リストを読み込む */
 if(NULL==(fp=fopen(inputlist, "r"))){
   printf("cannot open %s\n", inputlist);
   exit(1);
 }
 inimage=malloc(256*256*4); /* 入力画像のピクセル値(4byte)のためのメモリ確保 */
 outimage=malloc(outimagesizeX*outimagesizeY*4); /* 出力画像のピクセル値(4byte)
のためのメモリ確保 */
   /* 出力画像の各ピクセル値を 0 にリセット */
   for(i=0;i<outimagesizeX;i++){</pre>
   for(j=0;j<outimagesizeY;j++){</pre>
     outimage[i*outimagesizeY+j]=0.0;
   }
 }
 /* 構造の中にアウトプット WCS キーワードを置く */
 relax=1; /* 0は公式のWCSキーワードに定義されているFITSキーワードだけを認める。1は
0より緩い */
 ctrl =0; /* もし0なら、認められていないヘッダーカードを報告しない。値が1,2,3なら詳
細なエラーメッセージをこの順序で与える */
```

```
/* 出力用の FITS ファイルを作成する */
 if (fits_create_file(&fptr_out, outfile, &status)) {
   fits_report_error(stderr, status);
   return 1;
 }
 naxes[0] = outimagesizeX;
 naxes[1] = outimagesizeY;
 /* 用意した出力ファイルのヘッダーにキーワードを書き込む */
 fits_create_img(fptr_out, FLOAT_IMG, 2L, naxes, &status);
 /* 基本座標システムに FK5 を使用する */
 fits_write_key_str(fptr_out, "RADESYS", "FK5", "Coordinate system", &status);
 /* 2000年の春分点 (equinox) を利用する */
 fits_write_key_dbl(fptr_out, "EQUINOX", 2000.0, 3, "Equinox",&status);
 /* CTYPEに局所球面座標のタイプを入力している。PRJはプログラムを動かす時に入力するパ
ラメータで、AIT などの投影法を指定している */
 sprintf(RAproj, "RA---%s", PRJ);
 fits_write_key_str(fptr_out, "CTYPE1", RAproj, "DEC projection",&status);
 sprintf(DECproj, "DEC--%s", PRJ);
 fits_write_key_str(fptr_out, "CTYPE2", DECproj, "DEC projection",&status);
 /* 参照ピクセル座標を入力したパラメータの CRPIX1, CRPIX2 にする */
 fits_write_key_dbl(fptr_out, "CRPIX1", crpix1, 13, "X reference pixel",&status);
 fits_write_key_dbl(fptr_out, "CRPIX2", crpix2, 13, "Y reference pixel",&status);
 /* 座標中心位置の設定 */
 fits_write_key_dbl(fptr_out, "CRVAL1", x, 13, "RA of reference pixel",&status);
 fits_write_key_dbl(fptr_out, "CRVAL2", y, 13, "DEC of reference pixel",&status);
 /* 座標スケール値を入力パラメータの pixelsize の値にする */
 fits_write_key_dbl(fptr_out, "CDELT1", -pixelsize, 13, "X pixel increment
                   (degree)",&status);
 fits_write_key_dbl(fptr_out, "CDELT2", pixelsize, 13, "Y pixel increment
                   (degree)",&status);
 /* 出力ファイルに書き込んだヘッダーを読み込み○でないことを確認 */
 if (fits_hdr2str(fptr_out, 1, NULL, 0, &header, &ncards, &status)) {
   fits_report_error(stderr, status);
   return 1;
 }
```

```
/* 構造の外にアウトプット WCS キーワードを置く */
if (status = wcspih(header, ncards, relax, ctrl, &nreject, &nwcs, &wcs_out)) {
  fprintf(stderr, "wcspih ERROR %d: %s.\n", status, wcs_errmsg[status]);
if (status = wcsset(wcs_out)) {
 fprintf(stderr, "wcsset ERROR %d: %s.\n", status,
 wcs_errmsg[status]);
wcsprt(wcs_out);
free(header);
nfile=0;
while(fscanf(fp, "%s", infile)==1){
 nfile++;
  /* 入力 FITS ファイルを開く */
  if (fits_open_file(&fptr_in, infile, READONLY, &status)) {
   fits_report_error(stderr, status);
   return 1;
  }
  /* 全 FITS ファイルの CRVAL 読み込み */
 fits_read_keys_dbl(fptr_in, "CRVAL", 1, 2, crval, &num, &status);
  /* 範囲指定 */
  if(x-a < crval[0] \&\& crval[0] < x+b \&\& y-b < crval[1] \&\& crval[1] < y+b){
   printf("%s\nKeyword CRVAL1:%g\nKeyword CRVAL2:%g\n", infile, crval[0],
          crval[1]);
  /* 条件に合う入力ファイルのヘッダーを読み込む */
  if (fits_hdr2str(fptr_in, 1, NULL, 0, &header, &ncards, &status)) {
   fits_report_error(stderr, status);
   return 1;
 }
  /* 入力画像 (256x256) の各ピクセル値を 2 次元配列に読み込む */
 fits_read_2d_flt(fptr_in, 0, 0.0, 256, 256, 256, inimage, &anynul, &status);
  /* 構造の中にインプット WCS キーワードを置く */
  if (status = wcspih(header, ncards, relax, ctrl, &nreject, &nwcs, &wcs_in)) {
   fprintf(stderr, "wcspih ERROR %d: %s.\n", status, wcs_errmsg[status]);
  }
 free(header); /* 獲得したメモリの解放 */
  if (status = wcsset(wcs_in)) {
```

```
fprintf(stderr, "wcsset ERROR %d: %s.\n", status,
     wcs_errmsg[status]);
   }
   /* 入力画像 (256x256) を読み込み、元のピクセル座標を世界座標に変換する */
   for(i=0;i<256;i++){
     for(j=0;j<256;j++){
/* (1,1) が左下で X は右に行くほど増加し、Y は上にいくほど増加し、X=i+1,Y=j+1 と表せる時
のイメージ座標 (X,Y) を定める */
X_{in} = (float)j+1.0;
   Y_{in} = (float)i+1.0;
/* WCSLIB を使って World Coordinates(赤経と赤緯)を算出する */
pixcrd_in[0][0]=(double) X_in;
   pixcrd_in[0][1]=(double) Y_in;
wcsp2s(wcs_in,1,2,pixcrd_in[0],imgcrd[0],phi,theta,world[0],&status);
alpha = world[0][0];
   delta = world[0][1];
/* WCSLIB を使って (X,Y) 座標のピクセル値 (flux) を算出する */
wcss2p(wcs_out,1,2,world[0],phi,theta,imgcrd[0],pixcrd_out[0],&status);
X_out=(float) pixcrd_out[0][0];
   Y_out=(float) pixcrd_out[0][1];
i_out = floorf(X_out)-1;
   j_out = floor(Y_out)-1;
   /* 入力画像の (i,j) ピクセルの flux を対応する出力画像の (i_out, j_out) ピクセルに書
き込む */
if(i_out>=0&&i_out<outimagesizeX&&j_out>=0&&j_out<outimagesizeY) {</pre>
  outimage[i_out+j_out*outimagesizeX] = outimage[i_out+j_out*outimagesizeX]
              + inimage[i*256+j];
     }
       }
     }
   }
   /* 入力ファイルを閉じる */
   fits_close_file(fptr_in, &status);
   wcsfree(wcs_in);
  imagesize=outimagesizeX*outimagesizeY;
  fits_write_2d_flt(fptr_out, OL, outimagesizeX, outimagesizeX, outimagesizeY,
                   outimage, &status);
```

```
/* 出力ファイルを閉じる */
fits_close_file(fptr_out, &status);
wcsfree(wcs_out);
}
```

A.2 STIFF

STIFF[12] は専門的な FITS イメージをよりポピュラーな TIFF フォーマットに変換するプログラムである。入力するイメージが 1 つだとグレースケールを出力し、 3 つだと赤、緑、青のカラーイメージを出力する。

STIFF は TERRAPIX のサイトよりダウンロードすることができる。

実際に動かす場合は以下のようにコマンド入力する。

入力イメージが1つの場合:

% stiff 入力画像 -c configuration-file [-Parameter1 Value1] [-Parameter2 Value2]....

入力イメージが3つの場合:

% stiff 入力画像1(赤)入力画像(緑)入力画像(青)-c configuration-file [-Parameter1 Value1] [-Parameter2 Value2]....

謝辞

本研究を行うにあたり、データの提供をしてくださった宇宙科学研究本部の海老沢先生に感謝いたします。実際の「すざく」のプラネタリウム投影においても、丁寧なご説明をしていただきありがとうございました。

また、日常の議論を通じて多くの知識や示唆を頂いた宇宙情報解析研究室の皆様に感謝いたします。

そして最後になりましたが、研究全般において指導してくださった指導教官の久保田先生に深く感謝いたします。プラネタリウムの基礎知識から研究の進め方など全てにおいて大変お世話になりました。

皆様のおかげで無事に総合研究論文を完成させることができました。本当にありがとうございました。

関連図書

- [1] 西尾光史 総合研究論文(芝浦工業大学) 2007
- [2] FITS の手引き 第 5.1 版 監修 天文情報処理研究会、発行国立天文台天文学データ解析 計算センター
- [3] すざくヘルプ編「すざく解析マニュアル」
- [4] Mitsuda, K., et al. 2007, PASJ, 59, 1
- [5] Koyama, K., et al. 2007, PASJ, 59, 23
- [6] Kokubun, M., et al. 2007, PASJ, 59, 53
- [7] Takahashi, T., et al. 2007, PASJ, 59, 35
- [8] "Representations of world coordinates in FITS", (WCS Paper I)
 E.W. Greisen and M. Calabretta, Astron.& Astrophys., 395, 1061-1075, 2002
- [9] "Representations of celestial coordinates in FITS", (WCS Paper II)
 M. Calabretta and E.W. Greisen, Astron. & Astrophys., 395, 1077-1122, 2002
- [10] "Representations of spectral coordinates in FITS", (WCS Paper III)
 E.W. Greisen, M.R. Calabretta, F.G. Valdes, and S.L. Allen, Astron.& Astrophys., 446, 747-771, 2006
- [11] "Representations of distortions in FITS world coordinate system", (Paper IV) M.R. Calabretta et al., Apr. 22, 2004
- [12] STIFF v1.10 User's guide

表目次

2.1	すざくの特性 (すざくファーストステップガイドより)	4
2.2	検出器の性能	5
4.1	得たファイルの特徴・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	19

図目次

1.1	投影イメージ	1
2.1	すざく外観	3
2.2	XIS の外観	4
2.3	HXD の外観	5
3.1	単純な FITS ファイル構造	7
3.2	一般的な FITS ファイル構造	7
3.3	FITS ファイルの閲覧	8
3.4	ds9:使用例	9
3.5	Hammer-Aitoff 投影法	12
3.6	Gnomonic 投影法	13
3.7	Tangential spherical cube 投影法	14
3.8	Zenithal equal-area 投影法	15
3.9	Plate Carree 投影法	16
3.10	Polyconic 投影法	17
4.1	soft バンド [0.4-1.5keV]	19
4.2	medi バンド [1.5-4keV]	20
4.3	hard バンド [4-10keV]	20
4.4	・ 投影法 TSC での一台の画像 分	21
4.5	得られた全天画像	23
4.6	四隅の座標確認	24
5.1	単純な拡大画像 例:天体 SN1006	27
5.2	得られた拡大画像	31
5.3	log で表記	32
5.4	合成拡大画像 例:天体 SN1006	32
5.5	soft バンド+medi バンド	33
5.6	節囲選択	34
5.7	最終的な拡大画像 例:天体 SN1006	34
5.8	かに星雲....................................	35
5.9	はくちょう座 X-1	35
	SN 1572	36
-	いて座	36
5 12	プラネタリウム番組	37