

卒業論文

CABINを用いた
VR破面解析技術
1p ~ 71p 完

平成 13年 2月 9日 提出

指導教官 酒井 信介 教授

90262 山田 直秀

目次

第1章 序論	7
1.1 本研究の背景	8
1.1.1 従来のフラクトグラフィ解析	8
1.1.2 バーチャルリアリティ分野の発展	8
1.2 本研究の目的	10
1.3 本論文の構成	11
第2章 基礎理論	12
2.1 FRASTA 解析基礎	13
2.1.1 FRASTA の概念	13
2.1.2 従来の FRASTA の解析手順と実際の解析例	14
2.1.3 実際に解析を行なうときに生じる問題点	15
2.2 CABIN	17
2.2.1 CABIN の特徴	17
2.2.2 CABIN システム	17
2.2.3 CABIN の FRASTA 解析への応用	19
第3章 VR 用 FRASTA 解析プログラムの開発	20
3.1 解析手法の考案	21
3.2 VR 用 FRASTA 解析ソフトに求められる機能	23
3.3 CABIN 用 FRASTA 解析ソフトの開発	23
3.3.1 使用するインターフェイス	23
3.3.2 SEM によって取得したデータの読み取り	23
3.3.3 破面表示のためのアルゴリズム	25
3.3.4 破面の移動・回転のアルゴリズム	25
3.3.5 高さデータの比較のアルゴリズム	26
3.3.6 破壊面積の計算のアルゴリズム	28
3.4 開発ソフトの CABIN への適用	29
第4章 FRASTA 解析適用結果	30
4.1 疲労破壊試験に用いた試験片	31

4.2	疲労破壊試験	32
4.2.1	疲労破壊	32
4.2.2	試験片の作成	33
4.2.3	破面観察	34
4.2.4	表示・マッチング	35
4.2.5	破壊面積と破面間距離の関係	42
第5章	考察	44
5.1	VR 技術の有効性	45
5.2	FRASTA 解析	45
第6章	結論	48
6.1	結論	49
6.2	今後の課題	50
	参考文献	51
	あとがき	54
	付 録 A SEM の仕様	56
	付 録 B CABIN のスペック	63
	付 録 C 開発プログラム	66

図 目 次

2.1	FRASTA の概念図	13
2.2	FRASTA の解析手順	14
2.3	破壊が進行した領域の透視図 (FAPP) の例 [4]	15
2.4	破壊が進行した領域の横断面図 (XSP) の例 [4]	16
2.5	CABIN	17
2.6	CABIN の描画システム	18
3.1	考案した手法の流れ	21
3.2	SEM	22
3.3	破面の VRML 表示	24
3.4	FFT 後の破面の VRML 表示	24
3.5	頂点のポリゴン表示	26
3.6	CABIN 上でのマッチングの様子	29
4.1	疲労破壊試験に使用した CT 試験片	31
4.2	疲労破壊試験に使用した試験機	32
4.3	疲労き裂進展特性 [16]	33
4.4	疲労破壊の試験条件	34
4.5	疲労破壊試験によって得られた試験片 (x25)(試験片)	35
4.6	観察された上破面 (x300)(試験片)	35
4.7	観察された下破面 (x300)(試験片)	35
4.8	観察された上破面 (x25)(試験片)	36
4.9	観察された下破面 (x25)(試験片)	36
4.10	上破面の VR 表示 (試験片)	36
4.11	下破面の VR 表示 (試験片)	36
4.12	上下破面の対応付け (試験片)	37
4.13	上下破面の引き離し (試験片)	38
4.14	VR 表示した上破面 (試験片)	39
4.15	VR 表示した下破面 (試験片)	39
4.16	上下破面の対応付け (試験片)	40
4.17	上下破面の引き離し (試験片)	41

4.18	破壊面積と破面間距離の関係(試験片)	42
4.19	破壊面積と破面間距離の関係(試験片)	43
5.1	条件変化点の上下破面(試験)	46
5.2	条件変化点の比較(試験)	46
5.3	条件変化点での上下破面(試験)	47
5.4	条件変化点の比較(試験)	47
A.1	SEM	57
C.1	プログラムフロー	67

表 目 次

4.1	試験片 の疲労破壊試験条件	33
4.2	試験片 の疲労破壊試験条件	34
B.1	CG Rendering	64
B.2	共有メモリ	64
B.3	立体視	64
B.4	トラッキング	64
B.5	同期	65
B.6	入力デバイス	65

第1章 序論

1.1 本研究の背景

1.1.1 従来のフラクトグラフィ解析

材料、特に金属が破壊した表面、すなわち金属破断面には破壊の進行状況を示す特徴的な模様が多く残されている。それゆえに、対象とする金属材料の破壊機構の解明や破壊原因の推定は、その模様から情報を得ることによって行う事が多く、この手法をフラクトグラフィ (Fractography) と言う。

走査型電子顕微鏡 (Scanning Electron Microscope = SEM) の登場以後、その焦点深度が大きいこと、倍率可変が大きいことなどの特徴から、SEM を用いた破面観察はフラクトグラフィ解析の重要な手段となっている。[2]

SEM 像はそのままでは2次元の像でしかなく、画像の濃淡から対象の立体形状を推測することは容易ではない。SEM 像は白黒の濃淡の画像として出力されるが、この像は通常の写真のような照明を受けた濃淡像とはことなるため、SEM 像を正しく理解するためには知識や経験が必要となる。そのため観察対象である破面の表面立体形状を正確に把握するためには別の手段が必要である。従来からの方法として、ステレオ写真 (人間が両眼から見る場合のように同じ対象の異なる側面が写ってる2枚の写真) を撮影して、2枚の写真から立体視する方法は行われていたが、コンピュータを活用して2枚のステレオ写真から自動的に立体形状を算出する手法が開発され、また直接破面の立体形状を測定できる機器が登場するなど、デジタルデータとして定性的に観察することも可能となっている。

一般に破面観察は、定性的かつ2次元的な観察に終わることが多いが、立体的な観察によって破面に関してより多くの知見を得ることができる。[5] その1つの例がトポ1グラフィ手法 (FRActure-Surface Topography Analysis = FRASTA) と呼ばれている手法 [4, 3] であり、この手法を用いると、本来見ることでできない破壊進行過程を再現することができる。

FRASTA では、2つの共役な破面 (互いに向き合っている、同じ破壊を起こした破面) を正しく対応づけることが必要である。しかしこの作業には数学的にはっきりした指標がなく、解析者が経験的に行っているのが現状である。[12]

また、一般に対応づけの作業と破壊過程のシミュレーションの作業は独立に行われている。

1.1.2 バーチャルリアリティ分野の発展

近年のコンピュータの性能向上とその価格低下は著しく、コンピュータは科学的な研究や各種の統計処理、文書処理などに欠かすことのできない道具となっている。コンピュータが膨大な量の計算をすることが可能になり、その計算結果のデータを把握する作業や必要なデータだけを抽出するためのソフトウェア的な可視化の技術

やインターフェイスの分野が形成され、盛んに研究がなされている。

その中でもバーチャルリアリティ (Virtual Reality = VR:仮想現実) と呼ばれる分野が急速に発展してきている。VR 技術は対象とする人間に対しその人があたかもコンピュータの作り上げた世界に入り込んでいるかのような「没入感」を与えることができる。また、「没入感」を実現させるためには人間と機械 (コンピュータ) との間で「自然な」対話を行うことが要求される。そこで疑似3次元のコンピュータグラフィックス (Computer Graphics = CG)[9] や実画像などの表示技術を基本に、人間の各種の感覚に対して働きかける技術、及び、その逆に人間の動作等の能動的な要素を読みとる技術といったものが多数研究・開発されている。

本大学の施設である、IML(Intelligent Modeling Laboratory) のCABIN(Computer Augment Booth for Image Navigation) は VR を実現するための代表例である。

1.2 本研究の目的

本研究の目的を以下に挙げる.

FRASTA 解析に対して VR 技術が有効である可能性があることに着目し,

- フラクトグラフィ解析方法であるトポグラフィ解析手法 (FRASTA) にバーチャルリアリティ (VR) 技術を応用し, 新しい FRASTA 解析システムを構築する.
- 開発したシステムを実際の破面解析に適用して, 開発した解析システムの有効性を検証する.

1.3 本論文の構成

第 1 章 序論 では、本研究の背景について概説し、本研究の目的を示した。

第 2 章 基礎理論 では、従来より用いられている FRASTA 解析技術、本研究で用いた CABIN の特徴について簡単に説明する。

第 3 章 VR 用 FRASTA 解析プログラムの開発 では、本研究で考案した手法と、その手法を実現させるために本研究で開発したソフトとアルゴリズムについて説明する。

第 4 章 FRASTA 解析結果 では、考案した手法の有効性を検証するため、考案した手法を用いて FRASTA 解析を行なった結果を示す。

第 5 章 考察 では、考案した手法を用いて FRASTA 解析を行なった結果から本研究で開発したシステムの有効性を検証する。

第 6 章 結論 では、本研究を通して得られた結論を総括し、今後の課題について述べる。

第2章 基礎理論

本章では本研究で用いる解析技術である FRASTA 法について述べ、その問題点を考察する。また、VR を実現するために用いる CABIN と呼ばれる施設の特徴、システムについて述べ、FRASTA 解析に CABIN を用いることで可能となる事柄について考察する..

2.1 FRASTA 解析基礎

2.1.1 FRASTA の概念

FRASTA は FRActure-Surface Topography Analysis を意味する言葉であり，対応する破断面の立体形状を基に，破壊に関する知見を得るために行なわれるフラクトグラフィ解析の一手法である．

対応する破断面を解析するというコンセプトは，最後にはラプチャに至る微小破壊の過程を正確に特徴づけることに基づく．例えば，荷重を受けると構造物はまず弾性的に変形し，なお荷重を受け続けると，欠損や応力集中部から局所的な塑性変形を起こす．さらに荷重を受け続けると，塑性域内またはその付近に微小き裂が進展する．微小き裂の表面応力はゼロになるので，き裂が生じた時点の変形状態で表面より下の物質はすぐさま「固まって」しまい，それ以上の塑性変形は起こさない．加えられる荷重は，き裂先端の前面のまだ破断していない物質にかかり，破断するまで塑性変形を続ける（図 2.1）．

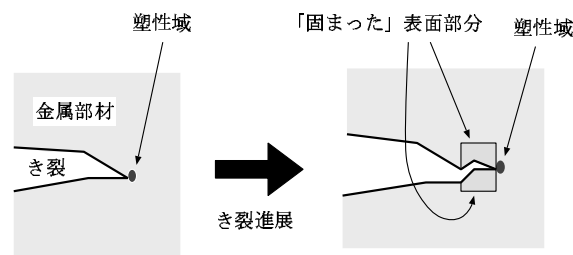


図 2.1: FRASTA の概念図

この一連の過程から，物質の局所的な塑性変形の大きさは，微小き裂の発生位置からの距離に応じて異なる．そして局所的な塑性変形の大きさが異なることから，破壊表面は不規則になる．これが破断面の高さ方向の凹凸であり，これらの凹凸は微小破壊の発生，進展の履歴を記録していることになる．

しかし，き裂それ自体が局所的な微小構造と干渉し，結果として生じたき裂進展の跡もまた，表面の不規則さをつくり出す．これら 2 つの表面の不規則さを分ける必要があるが，このことは対応する破断面（共役破面）を比較することで可能である．もしも不規則性が単にき裂先端と微小構造との干渉から生じ，破面は塑性変形していないのであれば，対応する表面の凹凸は正確に一致するはずである．一致しない場合は塑性変形があったことを示す．破壊する前に生じる局所的な塑性変形の大きさは，コンピュータを使って決定できる．この局所的な塑性変形の大きさの違いは，微小破壊の始めから終わりまでの一連の様子を再現してくれるのである．

このようにして，微小き裂の発生場所やき裂進展の様子がわかり，き裂進展速度と部材の変形の大きさとの関係や，き裂先端開口変位 (CTOD) などが測定できる．

さらに CTOD がわかることで、部材の破壊靱性値の値を知ることができる。

2.1.2 従来の FRASTA の解析手順と実際の解析例

FRASTA 法の解析手順の概要を図 2.2 に示す。一般的な FRASTA 法の解析に必要な手順は以下の通りである。

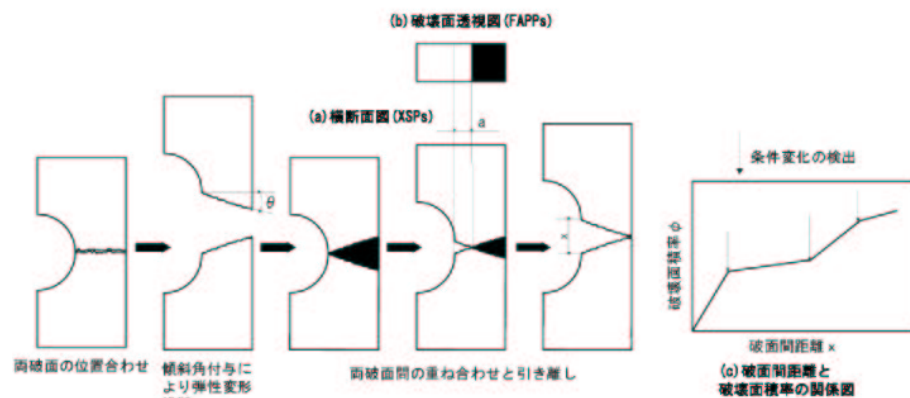


図 2.2: FRASTA の解析手順

1. 破面開放と切断

不完全分離破面に対しては、その破面に損傷や変形を与えないよう、低温脆性あるいは疲労破壊で分離する。また、大きな破面については、解析可能な寸法に切断する。

2. 破面凹凸形状の計測とデータ取り込み

例えば、SEM 等により破面凹凸を計測し、コンピュータにデータを取り込む。一度に計測できる範囲より大きい破面は分割計測を行なう。

3. 両破面の位置合わせ

コンピュータ上で両破面の位置合わせを行なう。

4. 弾性変形分の補正

実際にき裂が伝播する際には外力が作用しているため、破面には弾性変形による開口分が加わっている。これを模擬するために、き裂伝播方向に傾斜角 θ を与える。この傾斜角は、き裂前進先端の形状が実際の破面上のビーチマークや最終き裂の形状と一致するように、また、主き裂の先端に発生した微小き裂が後戻りしないように試行錯誤的に決定する。なお、解析中は傾斜角を一定に保

持する。

5. 両破面の重ね合わせと引き離し

両破面間のギャップが完全に消失する位置まで破面を重ね合わせた後、逆に両破面を徐々に引き離すことにより、き裂の伝播過程を再現する。この結果として、図 2.2 に示すように、(a) き裂の横断面図 (XSPs)、(b) 破壊面透視図 (FAPPs) および (c) 破面間距離と破壊面積率の関係図の 3 種類の定量的な情報が得られる。

6. 判定

前項で得られた情報から、き裂の伝播挙動を解釈する。これが FRASTA 法のキーポイントである。

FRASTA 法を用いた 3 次元破面解析のアウトプットとしては、(a) 破壊面の開放過程から、き裂の発生起点を推定すること、(b) 破面間距離と破壊面積率の傾きの変化点から何らかの事象変化があったことを推定することである。

さらに FRASTA では、き裂先端開口変位 CTOD が、従来手法である破面写真からのき裂先端部のストレッチゾーン長さの計測、ステレオ写真をもとにした CTOD を計測する方法に比べて、前述のように対応する両破面をコンピュータ上で仮想的に重ね合わせた後、徐々に引き離すため容易に計測できる。

FRASTA 解析の結果表示として標準的な、破壊が進行した領域の透視図 (FAPPs) の例と、破壊が進行した領域の横断面図 (XSPs) の例をここに挙げておく [4]。

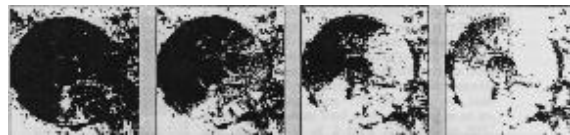


図 2.3: 破壊が進行した領域の透視図 (FAPP) の例 [4]

これまでに FRASTA により多くの研究が行なわれてきた。ステンレス鋼の高温曲げ疲労破面 [3]、ステンレス鋼の応力腐食割れ [4]、SiC/TiAl 複合材の引張り試験 [6]、ステンレス鋼の温度脆化 [7]、土壌過防食環境下の炭素鋼の水素誘起割れ [8]、破壊靱性試験 [10] など、その適用範囲は多岐にわたる。

2.1.3 実際に解析を行なうときに生じる問題点

実際に解析を実行するとき、共役破面の対応づけ作業に最も労力および時間を費やすことになる。解析の精度について考えた場合、従来から行なわれている FRASTA について、次の 2 点が問題となる。

1. 解析の精度は共役破面の対応づけの精度に大きく依存しているが、対応づけの判断にはっきりとした基準がないため曖昧さが残る。

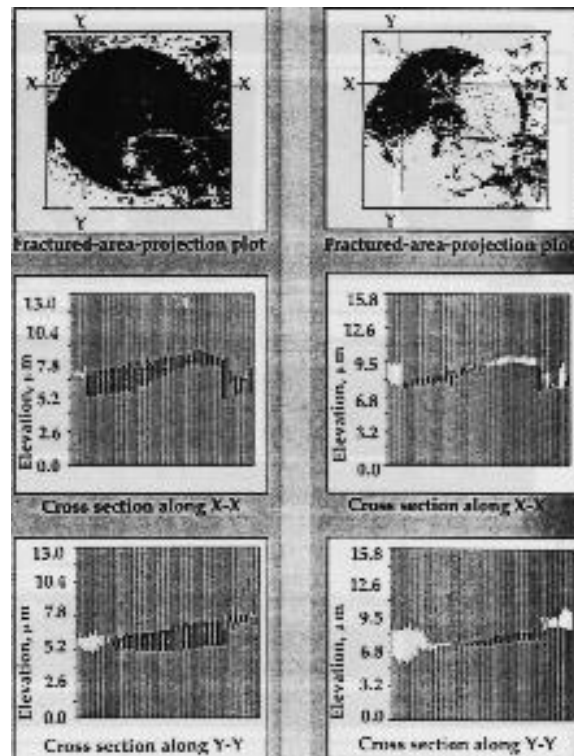


図 2.4: 破壊が進行した領域の横断面図 (XSP) の例 [4]

2. 破壊過程を詳細に解析するためには、微小破壊が起こっているすべての場所について各々対応をとる必要がある。一般に、対応づけの作業と破壊過程のシミュレーションの作業は独立に行なわれるが、これら2つの作業は同時に行なわれることが望まれる。その場合、対応をとる作業は大変繁雑になる。

2.2 CABIN

2.2.1 CABIN の特徴

CABIN(Computer Aided BIN)(図 2.5)とは東京大学IML(Intelligent Modeling Laboratory)内に設置されている没入型多面ディスプレイ装置である。上下左右前の5面に強化ガラスを設置し、装置内での立体視が可能となる。

CABINには以下のような特徴がある [14]。

- 高視野角による周辺視効果がある。
- 様々な入出力デバイスによるインタラクションが可能である。
- 液晶シャッターめがねによる立体視が可能である。
- 下面 15mm 厚二枚重ねの強化ガラスを用い、被験者が入り込むことが可能である。
- 視点連動により、様々な角度、位置からコンテンツを観察できる。

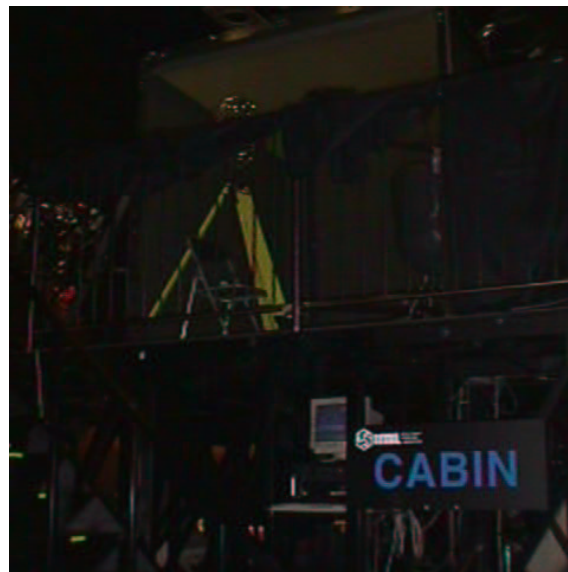


図 2.5: CABIN

2.2.2 CABIN システム

CABIN の描画システムを図 2.6 に示す。

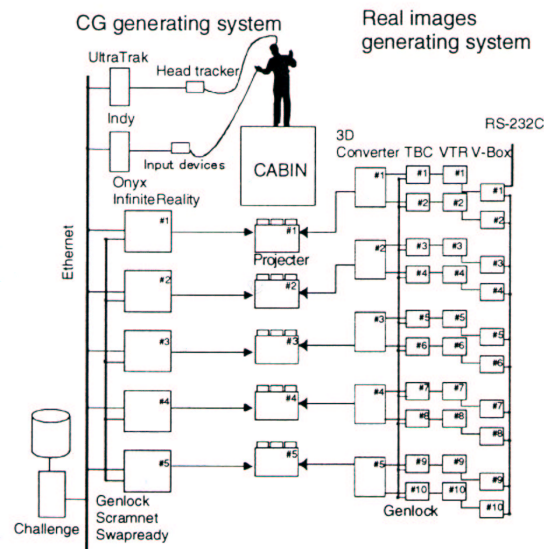


図 2.6: CABIN の描画システム

CG Rendering

5台のマシン (Front,Bottom,Top,Right,Left) はそれぞれ、5面のスクリーンに対応している。描画された映像はRGB信号でディスプレイに送られるが、それを分岐させてプロジェクターに信号を送っている。

共有メモリ

5台のマシン全てにSCRAMNetのボードが組み込まれており、どれか一つに書き込みが行なわれると、その情報を光ファイバーで他のSCRAMNetボード上のメモリに更新する。

立体視

立体視はCrystal Eye(TM)を用いている。Crystal Eyeは液晶シャッター方式の両眼視差立体眼鏡である。液晶シャッターの切り替えは、赤外線エミッタから出力される。

入力デバイス

Nintendo64コントローラや、振動ポインタなどが入力デバイスとして使用可能である。

トラッキング

三次元位置トラッキングは Polhemus 社 Ultratrak を用いている。Ultratrak は磁気式一錢さで磁場を生成するソース部分と、磁気をセンシングするセンサー部分から成り立つ。視点位置、手や足などの身体位置をトラッキングすることによってさまざまなインタラクションを可能にする。

同期

表示する物体の位置や座標系などの各面について共通なデータを共有メモリ内に置き、front をソースにして同期を行なうことによって仮想空間が生成される。

2.2.3 CABIN の FRASTA 解析への応用

CABIN は 3 次元の VR 表示が可能である。3 次元の VR を用いることによって高さ方向への広がり生まれ、従来の 2 次元の VR に比べ、より人間の各種感覚に訴えることが可能になることが期待できる。また、ディスプレイを見ながらの作業を要する 2 次元の VR 表示では作業が困難であるが、CABIN では液晶シャッター眼鏡による立体視によって、表示物体があたかも手元にあるような感覚や、没入感を得ることが可能になる。このことから FRASTA 解析の対応付けに CABIN を用いることでより精度よい対応付けが可能になると考えられる。

第3章 VR用FRASTA解析プログラムの開発

本章ではFRASTA解析をCABIN上で行なうために考案した手法について説明する。また、その手法を用いるために必要な機能を考察し、必要な機能を実現させるために本研究で開発したプログラムについて説明する。

3.1 解析手法の考案

2.2.1 節で述べた, CABIN の, 人間の感覚に鋭く訴えるための 3 次元空間を生成する特性が, FRASTA 解析において 2.1.3 節の問題点を軽減することに有効であると考え, 図 3.1 のような解析手法を考案した. SEM を用いて取得した高さデータを表示し, 没入空間内で解析者がコンピュータと対話形式で破面を動かし, マッチングした後, 破面間距離の算出などの計算を行なうという手法である.

1. SEM

SEM(図 3.2) を用いて破面を観察し, 高さデータを取得する.

2. CABIN

SEM を用いて取得した破面の高さデータをコンピュータに移し, CABIN 上で破面を表示する. 共役な 2 破面をそれぞれ独立に移動, 回転させ, 2 破面のマッチングを行なう. このとき 2 破面の特徴的な点を合わせるなどしてマッチングを行なう.

3. 解析

マッチング後, 一方の破面だけを高さ方向に動かして 2 破面の破面間距離を伸ばしていき, 破面間距離と破壊面積率を計算する.

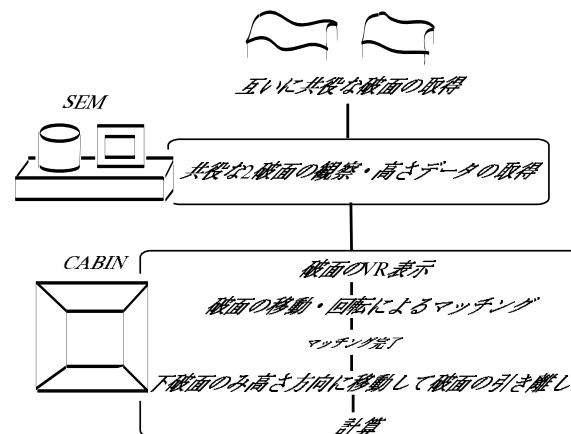


図 3.1: 考案した手法の流れ



图 3.2: SEM

3.2 VR用FRASTA解析ソフトに求められる機能

前節で考案した手法を用いてFRASTA解析を行なうためにはCABIN上で以下のような機能が必要であった。

- SEMを用いて取得した2破面の高さデータの読み取り
- 共役な2破面の表示
- コントローラ入力による2つの破面の独立移動，回転
- 2破面の移動距離，破面間距離と破壊面積の計算

以上のような機能を備えたVR用FRASTA解析プログラムを新たに開発した。次節からそのために用いたアルゴリズムについて説明する。

3.3 CABIN用FRASTA解析ソフトの開発

3.3.1 使用するインターフェイス

CABINでの表示にはOpenGL[9, 11, 15]による表示とVRML(Virtual Reality Modeling Language)[17]形式による表示という2つの方法を用いることができる。しかし，VRML形式ではSEMの特性上図3.3のように破面をVR表示した際に走査線方向とは垂直の方向に波上の縞模様が現れ，CABIN上で表示すると立体感を損なうものであった。この線はFFT(Fast Fourier Transfer=高速フーリエ変換)[13]によって図3.4のようにある程度消すことが可能であったが立体感を出すには十分と言えるものではなかった。また，マッチングをするためには2つの破面が互いに独立に移動・回転を行なえなければならないが，その機能はOpenGLを用いた方が実現が容易なこともあり，ソフト開発にはOpenGLを用いることとした。

3.3.2 SEMによって取得したデータの読み取り

SEMによって取得したデータを読み取る。読み取るデータは

- File 番号
- データタイプ
- 測定日時
- セーブ日時

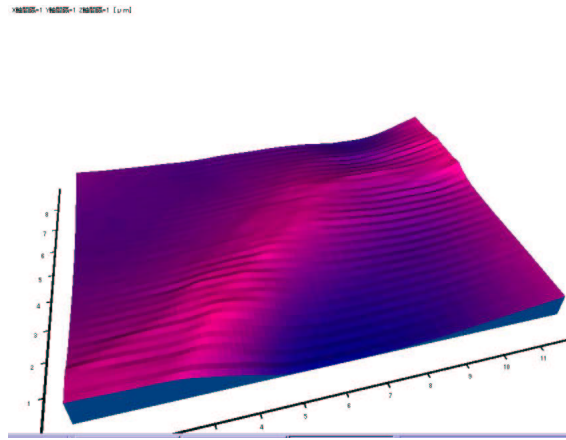


図 3.3: 破面の VRML 表示

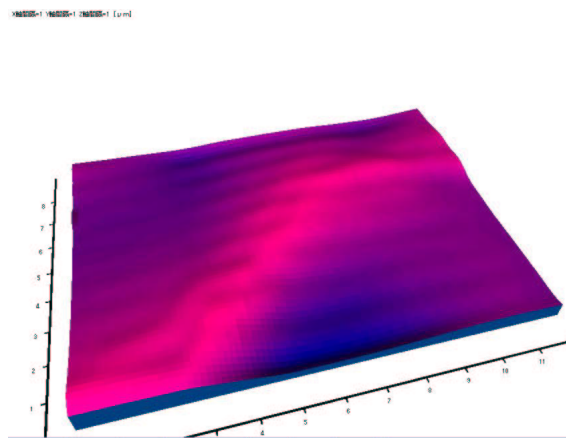


図 3.4: FFT 後の破面の VRML 表示

- 注釈
- 測定倍率
- 測定方向
- 測定スキャン本数
- 1 走査線上のサンプリング点数
- 測定開始位置
- 測定終了位置

- 補正データのサンプリング数
- x 方向データ間隔
- y 方向データ間隔
- 高さデータ

のデータとした。

3.3.3 破面表示のためのアルゴリズム

1つの面を作るには最低3個の点が必要であり、逆に3個の点が定義されれば一つの面を定義することができる。よって表示する破面は三角形のポリゴン表示とした。すなわち破面の x 方向 i 番目、 y 方向 j 番目の座標を $(x(i), y(j), z(i)(j))$ とし、図 3.5 のように点 $(x(i), y(j), z(i)(j))$ と点 $(x(i), y(j+1), z(i)(j+1))$ 、点 $(x(i+1), y(j), z(i+1)(j))$ の3点を結んだ三角形(図中)と点 $(x(i), y(j+1), z(i)(j+1))$ と点 $(x(i+1), y(j), z(i+1)(j))$ 、点 $(x(i+1), y(j+1), z(i+1)(j+1))$ の3点を結んだ三角形(図中)のポリゴンを作り表示した。

また、破面の色が単色であっては高さ情報を直感的に把握することは難しい。よって頂点の z 座標の違いによって色を変えて表示した。すなわち、頂点の色は z 座標の一番小さい頂点と一番大きい頂点の z 座標の差を d とし、点 $(x(i), y(j), z(i)(j))$ と z 座標の一番小さい頂点の z 座標の差を a としたとき、 $r = a/d$ ($0 \leq r \leq 1$) を変数として、

上破面は

赤成分を $1 - r^2$ 、緑成分を $1 - r(1 - r)$ 、青成分を $1 - (1 - r)^2$ 、
とする色で表し、

下破面は

赤成分を r^2 、緑成分を $r(1 - r)$ 、青成分を $(1 - r)^2$ 、
とする色で表した。

色は数字が大きい程その成分が強いことを示し、その数字は $0.0 \sim 1.0$ までの値をとる。また、赤成分 = 1.0 、緑成分 = 1.0 、青成分 = 1.0 のとき白色で、赤成分 = 0.0 、緑成分 = 0.0 青成分 = 0.0 のとき黒色を示す。

3.3.4 破面の移動・回転のアルゴリズム

破面の移動・回転には2つの共役な破面が独立に可動する必要がある。OpenGL ではそのためのアルゴリズムが用意されており、`glPushMatrix()`、`glPopMatrix()` というコマンドを用いた。このコマンドによって、移動・回転・表示を破面ごとに一つ

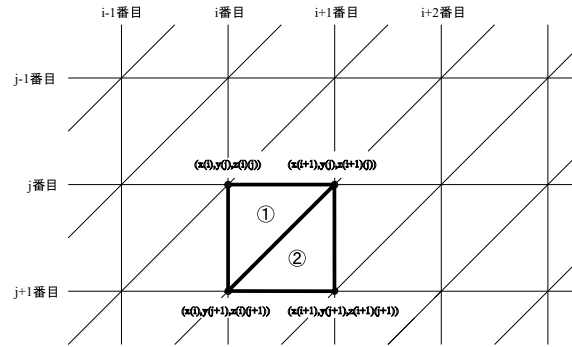


図 3.5: 頂点のポリゴン表示

のまとめとし、まず一方の破面を移動・回転・表示させ、その後視点のみ原点に戻し、他方の破面を移動・回転・表示させた。回転の中心は破面の中心とした。移動のためのコマンドは `glTranslate`、回転のためのコマンドは `glRotate` である。また、移動・回転させる破面をフラグによって制御して、ボタン操作で容易に、対象とする破面を変えることを可能にした。

2 破面が同時に移動・回転するときは視点の位置を移動させることとした。

これらの移動・回転は CABIN の入力デバイスの一つである Nintendo64 コントローラを用いて行なえるようにした。

3.3.5 高さデータの比較のアルゴリズム

表示した 2 破面の破壊面積を求めるためにはまず、表示している上下破面の位置関係を求めることが必要となる。下破面のある点が上破面の対応する点より下の位置にある場合のみそこが破壊されているとみなされるからである。また、2つの破面は独立に移動・回転するので下破面の x 方向 i 番目、 y 方向 j 番目のデータが上破面 x 方向 i 番目、 y 方向 j 番目のデータと比較すればよいというわけではない。以下はマッチングの際、上下破面が移動・回転したときの対応する点を求めるアルゴリズムである。

移動

上下破面の移動距離がデータ点に対して何個分であるかを計算する。つまり、上破面の移動距離を Xm_u 、 y 方向移動距離を Ym_u 、下破面の x 方向移動距離を Xm_d 、 y 方向移動距離を Ym_d 、上破面の x 方向データ間隔を $IntX_u$ 、 y 方向データ間隔を $IntY_u$ 、下は面の x 方向データ間隔を $IntX_d$ 、 y 方向データ間隔を $IntY_d$ とすれば上

下破面はそれぞれ,

$$X_u = Xm_u / IntX_u$$

$$Y_u = Ym_u / IntY_u$$

$$X_d = Xm_d / IntX_d$$

$$Y_d = Ym_d / IntY_d$$

により上破面はデータ点に対して x 方向に X_u , y 方向に Y_u , 下破面はデータ点に対して x 方向に X_d , y 方向に Y_d だけ移動したことになり, 下破面の点 $(x(i), y(j), z(i)(j))$ は上破面の点 $(x(i + X_d - X_u), y(j + Y_d - Y_u), z(i + X_d - X_u)(j + Y_d - Y_u))$ に対応することになる. この2点の高さ $z(i)(j)$ と $z(i + X_d - X_u)(j + Y_d - Y_u)$ を比較すればよいことになる.

回転

z 軸回転 回転後の x, y 座標が一致する点を求め, その2点の差がデータ点に対して何個分であるかを計算する. つまり, 上破面が角度 θ だけ回転したとき, 回転中心から距離が x 方向に lx_u , y 方向に ly_u だけ離れた点は,

$$LX_u = lx_u \cos \theta - ly_u \sin \theta$$

$$LY_u = lx_u \sin \theta + ly_u \cos \theta$$

より, 回転中心から距離が x 方向に LX_u , y 方向に LY_u だけ離れた点に移動する.

同様に下破面のが角度 ϕ だけ回転したとき, 回転中心から距離が x 方向に lx_d , y 方向に ly_d だけ離れていた点は x 方向に LX_d , y 方向に LY_d だけ回転中心から離れた点に移動する. ここで,

$$LX_d = lx_d \cos \phi - ly_d \sin \phi$$

$$LY_d = lx_d \sin \phi + ly_d \cos \phi$$

である. これらが一致する, つまり, 回転中心の x, y 座標が同じであれば, $LX_u = LX_d$, $LY_u = LY_d$ なので,

$$lx_u \cos \theta - ly_u \sin \theta = lx_d \cos \phi - ly_d \sin \phi$$

$$lx_u \sin \theta + ly_u \cos \theta = lx_d \sin \phi + ly_d \cos \phi$$

となり,

$$lx_u - lx_d = (\cos \theta \cos \phi + \sin \theta \sin \phi - 1)lx_d + (\sin \theta \cos \phi - \cos \theta \sin \phi)ly_d$$

$$ly_u - ly_d = (\cos\theta\sin\phi - \sin\theta\cos\phi)lx_d + (\cos\theta\cos\phi + \sin\theta\sin\phi - 1)ly_d$$

と変形できる．つまり，上破面が角度，下破面が角度だけ回転したとき，回転中心の x, y 座標が同じであれば回転中心から (lx_d, ly_d) だけ離れた下破面の点 $(x(i), y(j))$ は上破面の点 $(x(i + \frac{lx_u - lx_d}{IntX}), y(j + \frac{ly_u - ly_d}{IntY}))$ と一致する．この2点の高さデータを比較すればよい．

$IntX, IntY$ はそれぞれ， x 方向データ間隔と y 方向データ間隔であり，ここでは上下破面ともその値は同じであるとしている．

3.3.6 破壊面積の計算のアルゴリズム

下破面の点 $(x(i), y(j), z(i)(j))$ が破壊状態にある (対応する上破面の点より z 座標が小さい) とき，表示時と同様に点 $(x(i), y(j), z(i)(j))$ と点 $(x(i), y(j+1), z(i)(j+1))$ ，点 $(x(i+1), y(j), z(i+1)(j))$ の3点を結んだ三角形と点 $(x(i), y(j+1), z(i)(j+1))$ と点 $(x(i+1), y(j), z(i+1)(j))$ ，点 $(x(i+1), y(j+1), z(i+1)(j+1))$ の3点を結んだ2つの三角形を作り，それぞれの三角形の面積を求める．破壊状態にある全ての点において面積を求め，その和を破壊面積とした．

三角形の一つの頂点から出ている2本のベクトルを $\vec{a}$, $\vec{b}$ とするとき，三角形の面積 S は

$$S = \frac{\det(\vec{a} \times \vec{b})}{2}$$

で求められる．

3.4 開発ソフトのCABINへの適用

以上のようなアルゴリズムを用いて開発したプログラムを用いた結果 CABIN 上で2つの共役な破面を表示し，独立に移動・回転させることが可能となった．また，CABIN の視覚効果により，共役破面のマッチング作業が容易かつ精度よいものとなった．図 3.6 は CABIN 上で共役破面を表示し，マッチング作業をしている様子である．

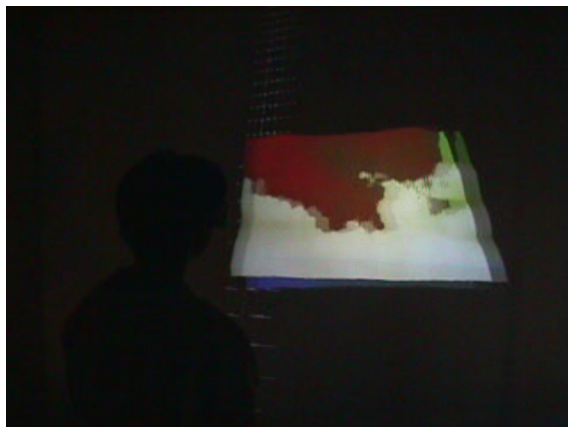


図 3.6: CABIN 上でのマッチングの様子

第4章 FRASTA解析適用結果

本章では前章で考案した手法の有効性を検証するため，実験破面を作成し，考案したFRASTA解析手法を用いてFRASTA解析を行なった結果について示す．

試験中に試験条件 (荷重など) を変えると破面上にビーチマークと呼ばれる一本の線が現れる。また FRASTA 解析において破面間距離と破壊面積の関係をグラフで表すと一定条件下ではそのグラフは直線状に伸びていくが試験条件が変化するとその点で直線の傾きが変化する。この理論を用いて本研究で考案した手法の有効性の検証を行なう。すなわち、疲労破壊試験中に条件 (応力拡大係数) を変化させビーチマークを出現させる。また FRASTA 解析において破面間距離と破壊面積の関係をグラフに表し、傾きの変化する点を導出し、その点でのき裂の位置導出する。その後、実際の破面のビーチマークと VR 上のき裂の位置を比較することを目的として疲労破壊試験を行ない、破面の FRASTA 解析を行なった。

4.1 疲労破壊試験に用いた試験片

図 4.1 に示すようなコンパクト引張標準試験片 (CT 試験片) を用いて条件を変え、2 種類の疲労破壊試験を行なった。CT 試験片の材質は高力 Al 合金 2024-T3 を用いた。疲労破壊に用いた試験機を図 4.3 に示す。

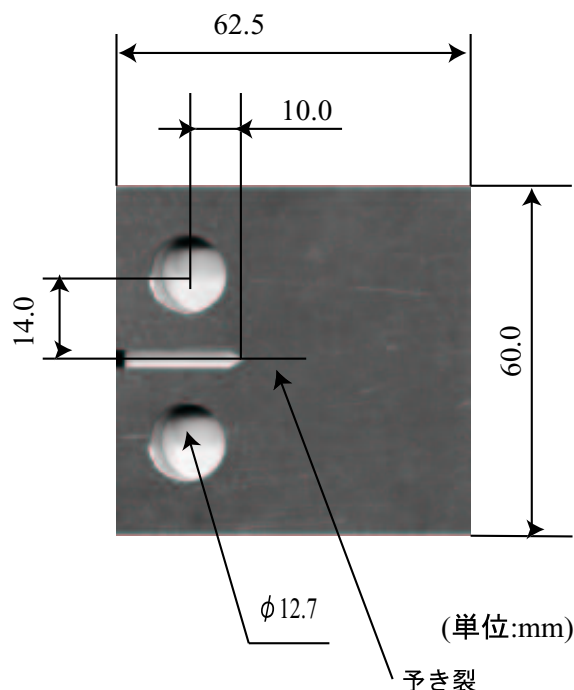


図 4.1: 疲労破壊試験に使用した CT 試験片



図 4.2: 疲労破壊試験に使用した試験機

4.2 疲労破壊試験

4.2.1 疲労破壊

疲労き裂進展は繰り返し数 N 依存形で，経験則から以下のような式，

$$\frac{da}{dN} = C\Delta K^m$$

で表される．この関数形は Paris 則と言われ， a はき裂長さ， ΔK は応力拡大係数の変動範囲を示す．また， m と C は材料定数である．

$\frac{da}{dN}$ と ΔK の間には図 4.3[16] で示すような関係があり，材料定数 m ， C が決定すれば ΔK から $\frac{da}{dN}$ を求めることができ，き裂進展までのおおよその時間を決定することができる．

FRASTA 解析の有効性を検証するため条件を変えた 2 種類の疲労破壊試験を行った．

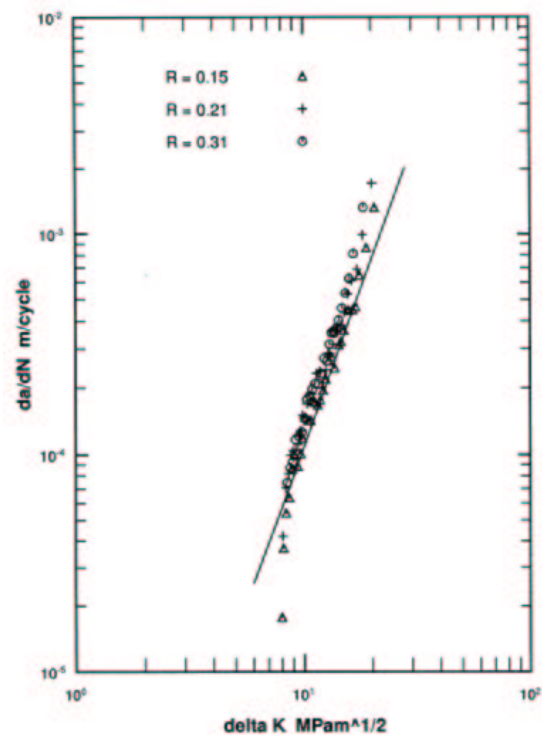


図 4.3: 疲労き裂進展特性 [16]

4.2.2 試験片の作成

試験片

き裂進展方向を制御するため 5mm の長さの予き裂を入れ、き裂長さを 15mm とした。試験条件は表 4.1 に示すように亀裂進展 3.2mm のところで変化させ、最終的にそのまま破壊した。疲労破壊試験中振動数は 10(HZ) であった。

疲労破壊試験		
条件	$K_{max}(\text{MPa}\sqrt{\text{m}})$	$K_{min}(\text{MPa}\sqrt{\text{m}})$
開始時	33.33	3.33
3.2mm き裂進展後	55.37	24.61

表 4.1: 試験片 の疲労破壊試験条件

試験片

破壊進行方向を制御するため 10mm の長さの予き裂を入れ，き裂長さを 20mm とした．表 4.2 のような条件のもと疲労破壊試験を行なった．条件変化はき裂進展 1mm のところで行なった．試験中振動数は 10(HZ) であった．

疲労破壊試験		
条件	$K_{max}(\text{MPa}M^{\frac{1}{2}})$	$K_{min}(\text{MPa}M^{\frac{1}{2}})$
開始時	38.52	6.42
1.0mm き裂進展後	64.20	32.10

表 4.2: 試験片 の疲労破壊試験条件

試験片 も試験片 も図 4.4 のように破壊条件変化前後では一定の応力拡大係数の振幅で試験した．

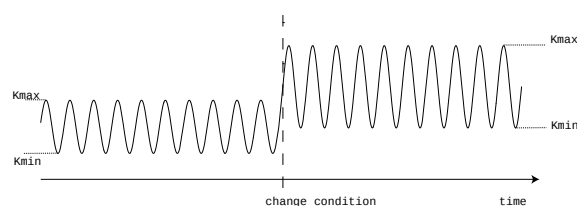


図 4.4: 疲労破壊の試験条件

4.2.3 破面観察

試験片

試験片 を走査型電子顕微鏡で観察すると図 4.5 のように観察された．このときの倍率は 25 倍であった．図中の線 L のように条件変化の点でビーチマークが現れた．

また，300 倍の倍率で観察すると上下の破面はそれぞれ，図 4.6，図 4.7 のように観察され，このときの高さデータを取得した．図 4.5 のビーチマークの線は図中で線 L' の位置に見られた．

試験片

SEM を用いて疲労破壊試験後の破面を 25 倍で観察した．それぞれ，図 4.8 と図 4.9 のように観察された．試験片 の時と同様に，条件変化の位置，すなわち線 M の位置に一本のビーチマークが観察できた．

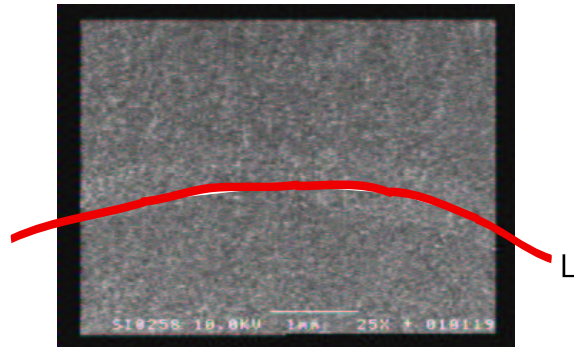


図 4.5: 疲労破壊試験によって得られた試験片 (x25)(試験片)

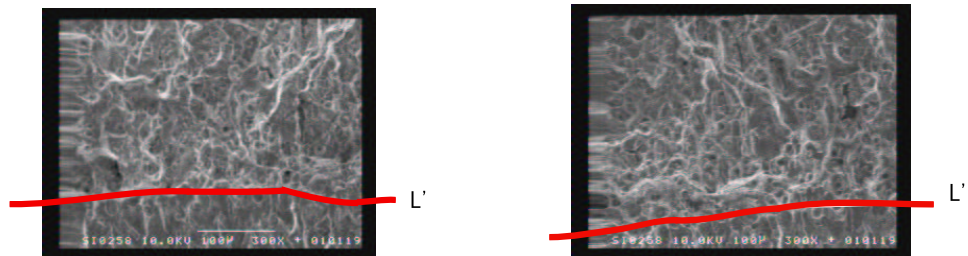


図 4.6: 観察された上破面 (x300)(試験片) 図 4.7: 観察された下破面 (x300)(試験片)

4.2.4 表示・マッチング

試験片

4.2 節の疲労破壊試験 で取得した破面の高さデータをもとに、CABIN 上で破面の VR 表示を行なった。破面の高さデータは SEM により 300 倍で観察したものから取得し、高さデータの数値は 160×120 であった。上下破面を VR 表示するとそれぞれ図 4.10, 図 4.11 のように、SEM の特性により、破面の端が歪んで観測された。

2つの破面を表示し、両破面の対応付けと引き離しを行なった。図 4.12 は対応付けの様子であり、ビーチマークや特徴的な点を重ね合わせることによってマッチングを行なった。図 4.13 はマッチング後上下破面を引き離していく作業での 2 破面の様子を示している。図中、赤色の破面が下破面、緑色の破面が上破面であり、両破面の境界がき裂進展位置を示す。

対応付けでは大まかな破壊の進行方向は既知のものとする。まず上破面のみを x 方向に動かし特徴的な点 (a) を重ね合わせた (図中 1~5)。次に視点を移動させ両破面を見やすい位置に移動・回転させ (図中 6), 続いて上破面のみを y 方向に移動させビーチマークを大まかに対応させた (図中 7)。最後に上破面のみを xy 平面内で回

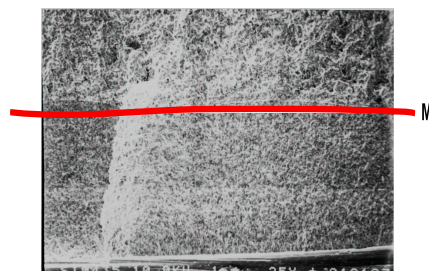
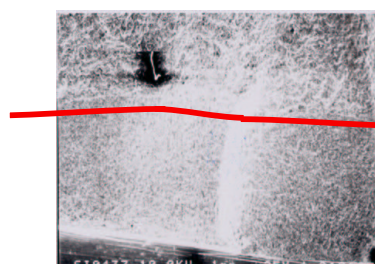


図 4.8: 観察された上破面 (x25)(試験片) 図 4.9: 観察された下破面 (x25)(試験片)

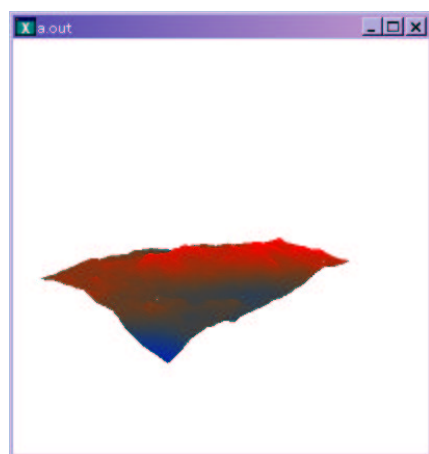
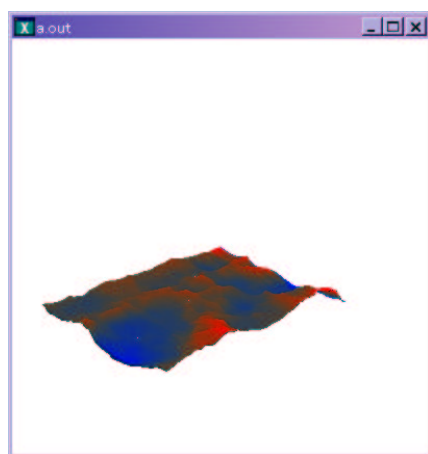


図 4.10: 上破面の VR 表示 (試験片) 図 4.11: 下破面の VR 表示 (試験片)

転させて両破面を対応付けた (図中 8).

続いて, 破壊進行過程が視覚的に把握できる位置に視点を移動し, 破面の引き離しを行ない破壊の進行過程を再現した.. 破面の引き離しは上破面のみを z 方向に動かしていき両破面の破面間距離を伸ばしていった.

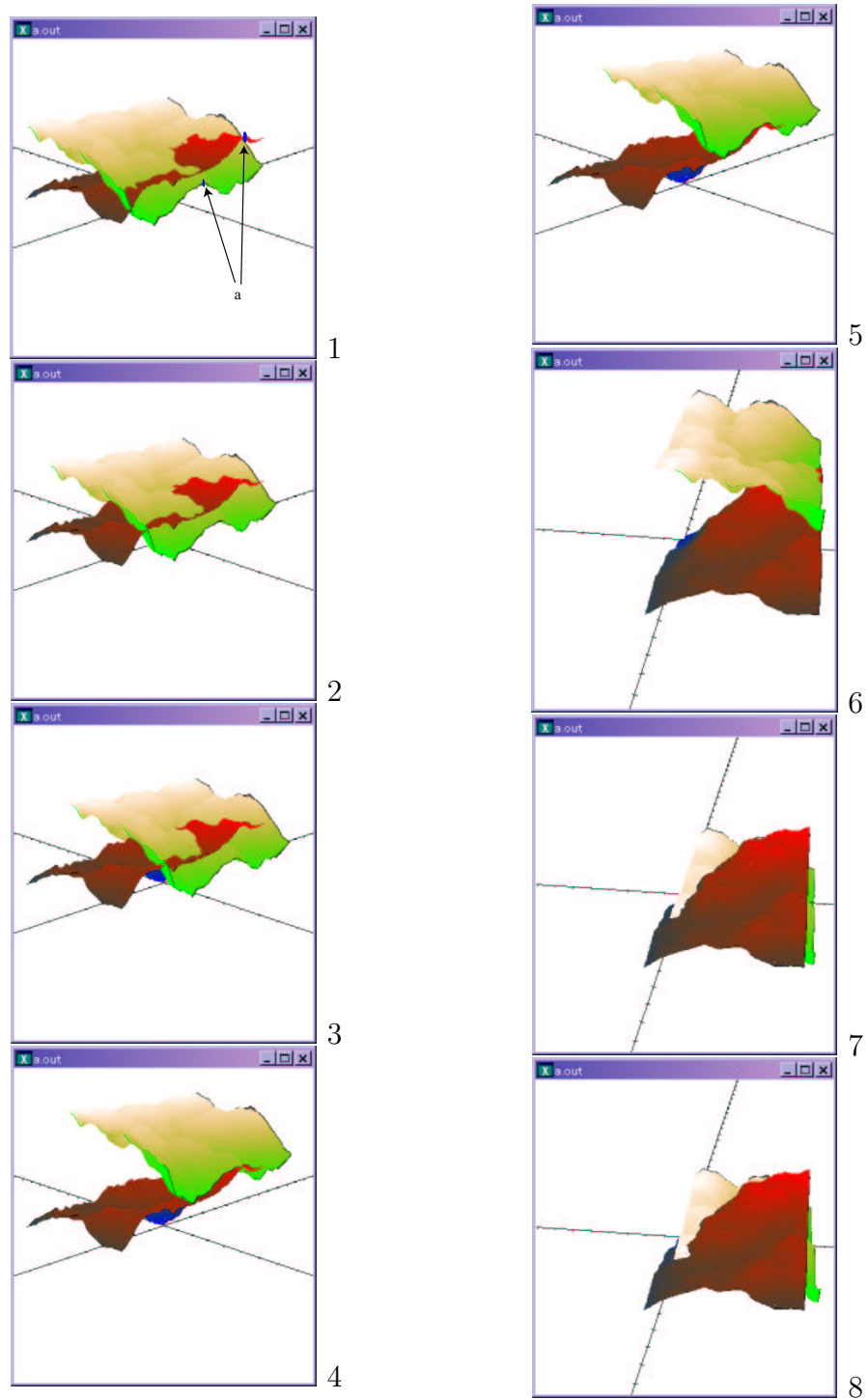


図 4.12: 上下破面の対応付け (試験片)

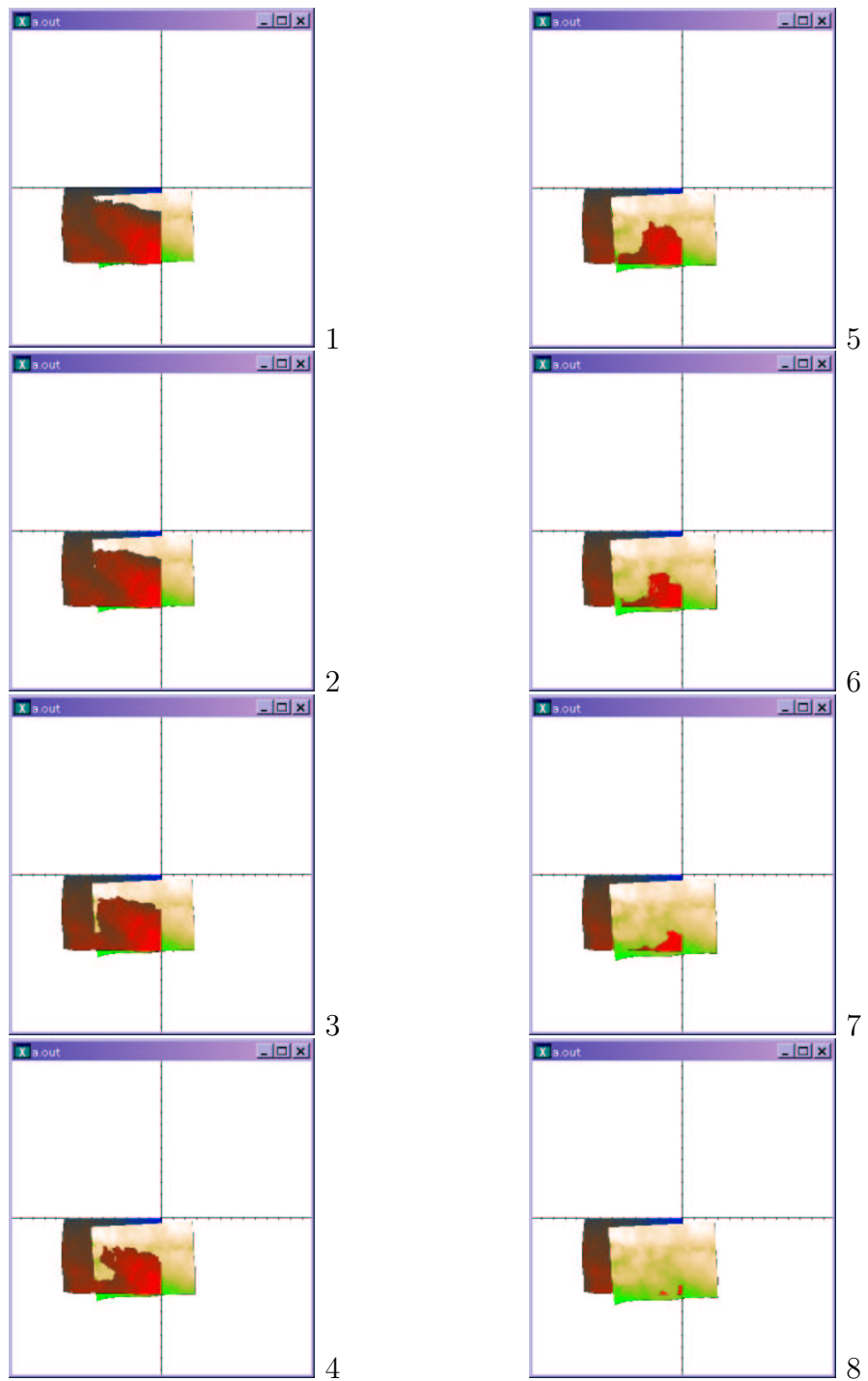


図 4.13: 上下破面の引き離し (試験片)

試験片

4.2.2 節の疲労破壊試験 で取得した破面の高さデータを，SEM により 25 倍の倍率で観察した際取得し，その高さデータを用いて CABIN 上で破面の VR 表示を行った．このとき高さデータの数値は 160×120 であった．上下破面を VR 表示するとそれぞれ，図 4.14，図 4.15 のようであった．試験片 同様，SEM の特性により，破面の端の方が歪んで観測された．

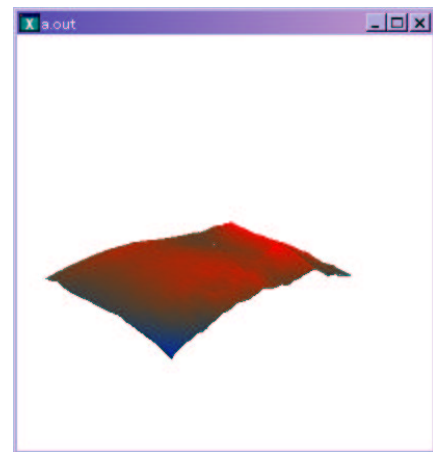
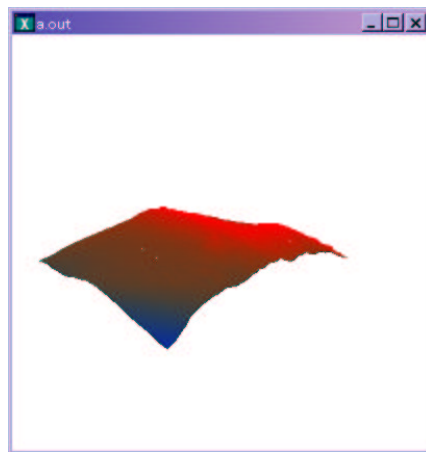


図 4.14: VR 表示した上破面 (試験片) 図 4.15: VR 表示した下破面 (試験片)

2つの破面を表示し，両破面の対応付けと引き離しを行なった．図 4.16 は対応付けの様子であり，図 4.17 はマッチング後上下破面の引き離しの様子である．図中，赤色の破面が下破面，緑色の破面が上破面であり，両破面の境界がき裂進展位置を示す．

対応付けでは試験片 同様に上破面のみを x 方向に移動させ，特徴的な点 (b) を重ねあわせ (図中 1～3)，次に両破面が見えやすい位置に視点を移動させ (図中 4)，続いて上破面のみを y 方向に移動させてピーチマークを大まかに対応付けた (図中 5)．最後に上破面のみを xy 平面内で回転させ (図中 6)，両破面の対応付けを行なった．

続いて，破壊進行過程が視覚的に把握できる位置に視点を移動し，破面の引き離しを行ない破壊の進行過程を再現した．破面の引き離しは上破面のみを z 方向に動かしていき両破面の破面間距離を伸ばしていった．

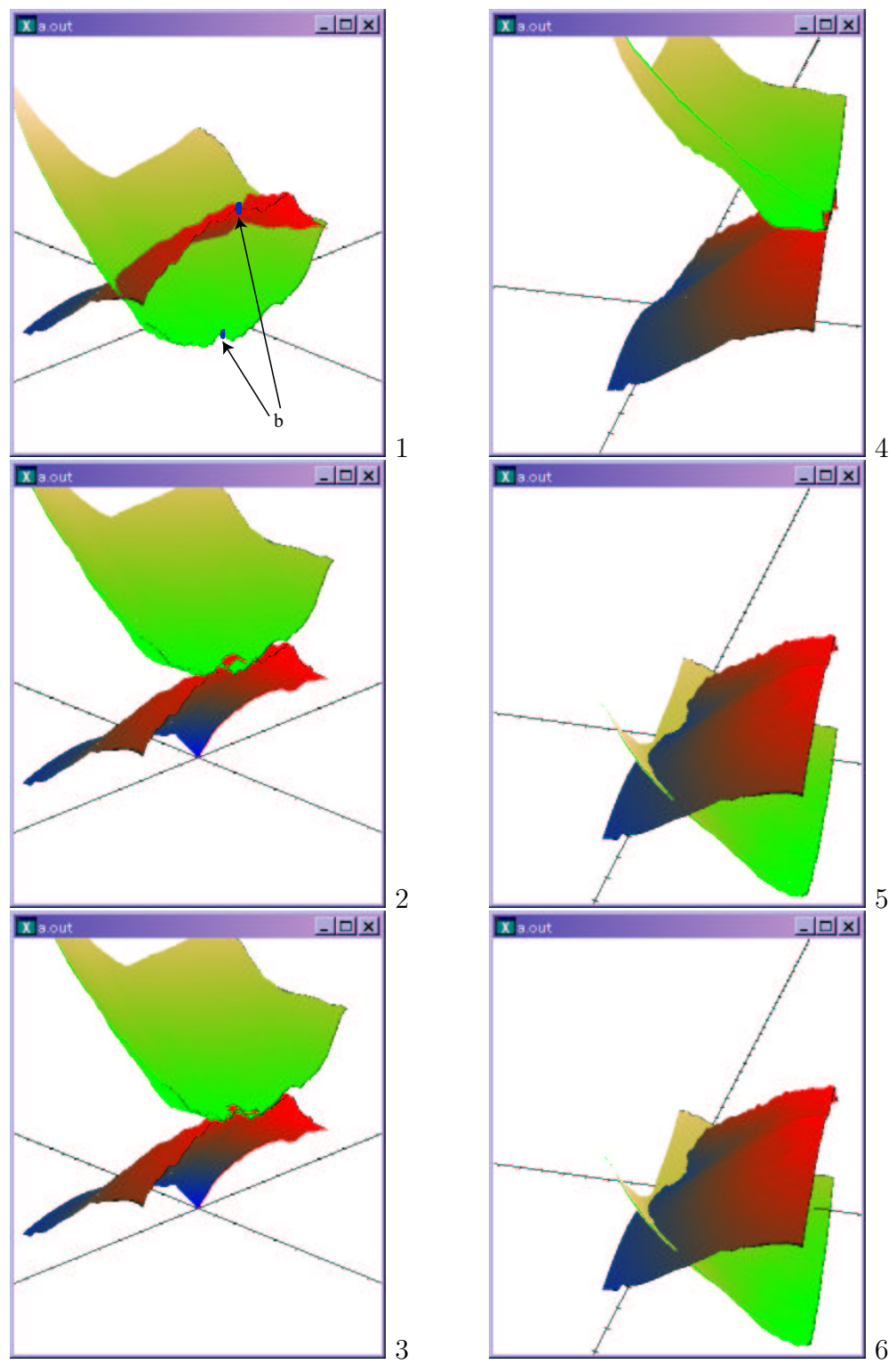


図 4.16: 上下破面の対応付け (試験片)

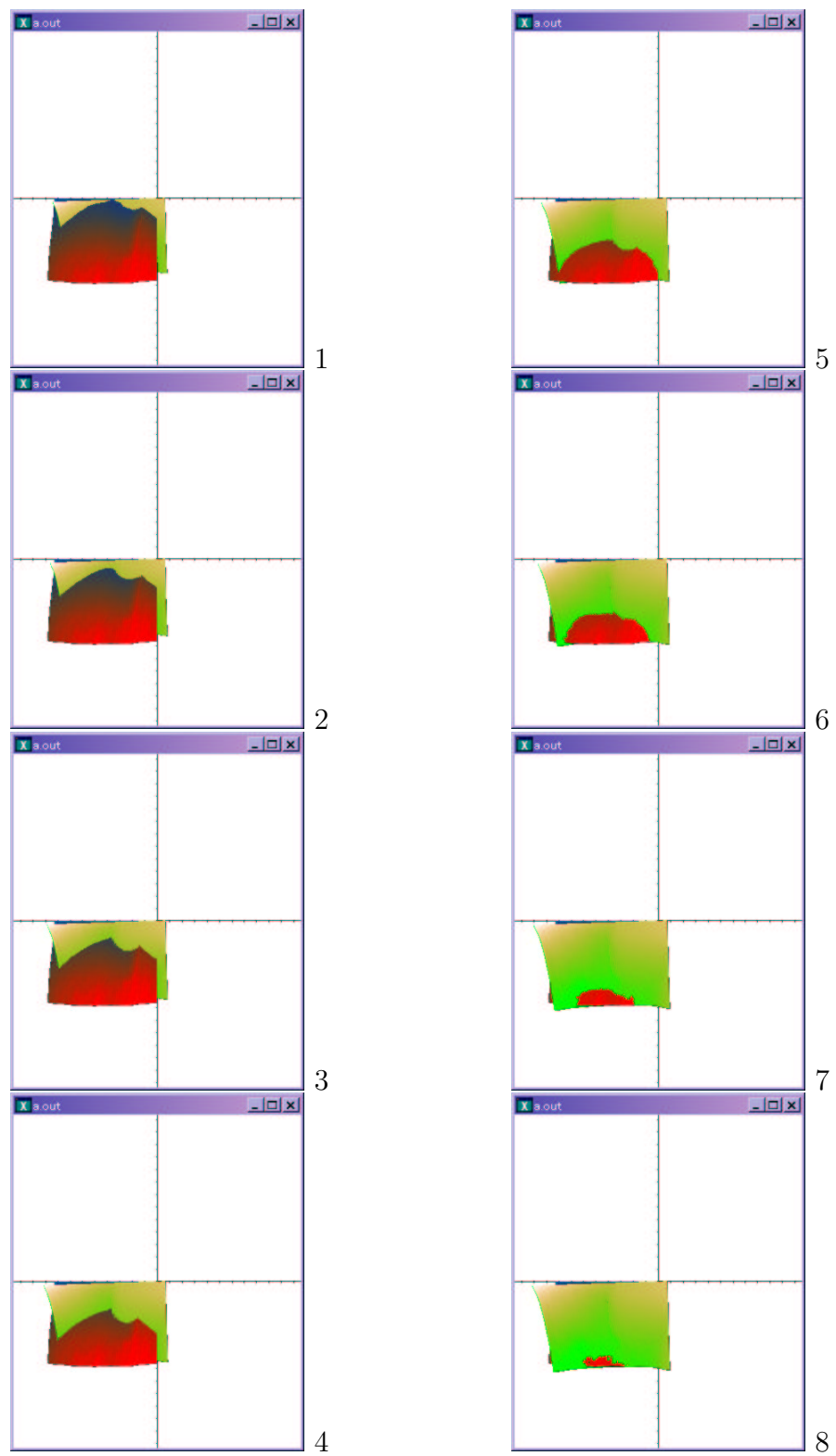


図 4.17: 上下破面の引き離し (試験片)

4.2.5 破壊面積と破面間距離の関係

試験片

破壊面積を求めた。破壊面積は3.3.6で示したように上破面上の点のうち、対応する下破面の点よりも z 座標が大きい点の周りのポリゴンの三角形の面積の和である。破面間距離と破壊面積の関係をグラフにすると図4.18のようになった。ここで破面間距離は破面間距離を伸ばしていく作業の開始点を0とし、その点から上破面が z 方向に移動した距離で表した。FRASTA解析の定義より、図4.18のグラフにおいて直線の傾きが変化するA点が、条件変化の点であると考えられる。

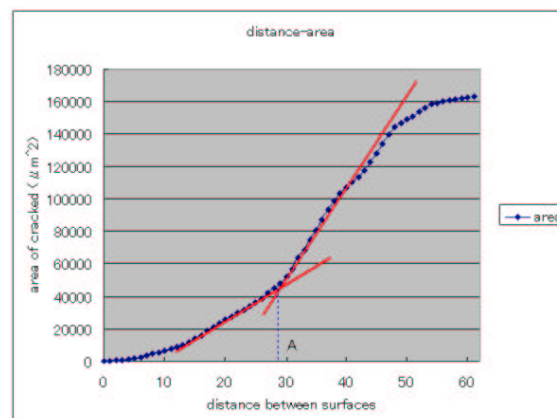


図 4.18: 破壊面積と破面間距離の関係 (試験片)

試験片

3.3.6 で示したように上破面上の点の中で対応する下破面の点より z 座標の大きい点の周りのポリゴンの三角形の面積の和で表した．破面間距離と破壊面積の関係を図に表すと図 4.19 のようになった．ここで破面間距離は 2 破面の引き離し開始点を 0 とし，上破面が z 方向に移動した距離で表した．FRASTA 解析の定義より，グラフの傾きの変化から B 点で条件変化が起こったものと考えられる．

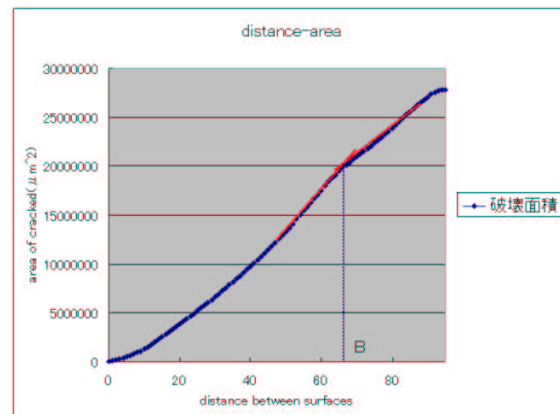


図 4.19: 破壊面積と破面間距離の関係 (試験片)

第5章 考察

本章では本研究で考案した手法を用いて FRASTA 解析を行なった結果より破壊条件変化点を導出し，実際の破面の条件変化点と比較して本研究で考案した手法の有効性を検証する．

5.1 VR 技術の有効性

FRASTA 解析に VR 技術を用いることによって両破面のマッチングが容易かつ精度のよいものとなった。これは VR 技術が以下のような特徴をもつためであると考えられる。

- 破面上の凹凸をより視覚的に把握することが可能である。
- 破面の微小部分にもぐりこむことが可能となり、微小部分の破壊過程を視覚的に把握することが可能である。
- 破面を多方向から見る事が可能である。
- 計算機上で数値を用いたマッチングよりも視覚的かつ直感的にマッチング作業が可能であるのでマッチングの時間が短縮できる。
- 破面の移動・回転に関して対話性・操作性の向上。
- 没入空間での作業において破面が手元にあるような感覚を得ることが可能である。

5.2 FRASTA 解析

試験破面 において図 4.18 の A 点での上下破面の位置関係は図 5.1 のようになり、き裂の位置と破面のビーチマークの位置を比較すると図 5.2 のようになる。図中の線 L1 は下破面の SEM 写真 (左) のビーチマークを表す線で下破面の大きさを VR 表示破面 (右) と合わせている。図 5.1 より両破面の端の位置でずれが見られる。

また、試験破面 において図 4.19 の B 点での上下破面の位置関係は図 5.3 のようになり、き裂の位置と破面のビーチマークの位置を比較すると図 5.4 のようになる。図中の線 L2 は下破面の SEM 写真 (左) のビーチマークを表す線で下破面の大きさを VR 表示破面 (右) と合わせている。図 5.3 より試験破面 同様、破面の端の位置でずれが見られる。

これら 2 つの解析の結果と実際の条件変化の点 (ビーチマーク) を比較するとずれが見られるが、これは以下の要因によるものと考えられる。

1. SEM 計測
2. 破壊面積の計算

1 は破面を SEM で観測すると破面の端の部分が歪んで観測されてしまうという SEM の特徴に起因する。2 は破壊面積計算時にき裂境界上の面積もポリゴンの三角形の面積で計算してしまい、実際の破壊面積よりも大きな値で計算してしまっていることによる誤差である。

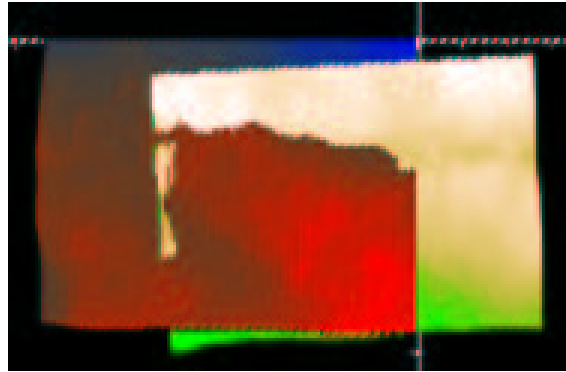


図 5.1: 条件変化点の上下破面 (試験)

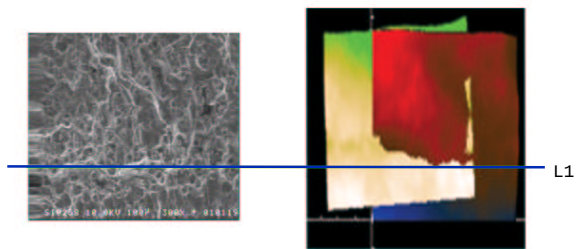


図 5.2: 条件変化点の比較 (試験)

図 5.2, 図 5.4 を見るとどちらも端の方がずれが大きい, このことから, 上の 2 つの原因において 1 の方が誤差の起因になっていると考えられる. 以上の考察より考案した手法は FRASTA 解析に有効であると考えられる.

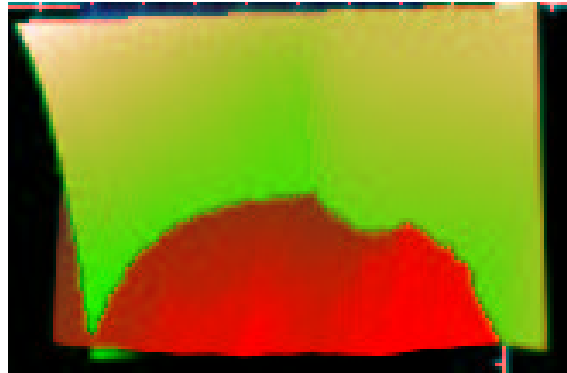


図 5.3: 条件変化点での上下破面 (試験)

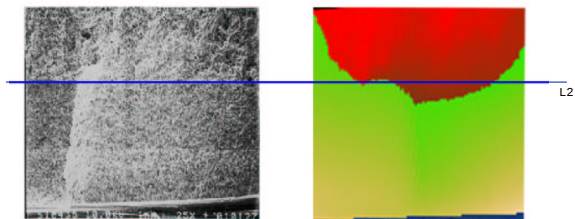


図 5.4: 条件変化点の比較 (試験)

第6章 結論

本章では前章の考察から導かれる結論を示し，今後 FRASTA 解析手法の精度を上げるための課題を考察する．

6.1 結論

本研究での結論は以下の通りである。

1. フラクトグラフィ解析手法であるトポグラフィ手法 (FRASTA) に CABIN を用い、新たに解析手法を考案した。
2. CABIN を用いることで、微小空間に没入し、仮想の破面を自由に操作することが可能になった。そして、立体形状で表される破面の任意の箇所を、様々な視点から自由に観察することが可能になった。
3. 開発したプログラムを用いて CABIN 上で破面を表示することにより、破面の凹凸感をより視覚に訴えることが可能になり、マッチング作業が容易かつ精度のよいものとなった。
4. 考案した手法により、従来の FRASTA 解析技術の問題点を改善することができた。すなわち、
 - 共役な 2 破面の対応付けの作業が容易、かつ精度の良いものとなった。
 - 破面の対応付けと破壊過程のシミュレーションが同時に行なうことが可能になった。
5. 開発した手法を、実際の破面に適用し、FRASTA 解析を行なった。本研究では条件変化点を求めた。その結果本研究で開発した手法は FRASTA 解析に有効であることが分かった。

6.2 今後の課題

FRASTA 解析の結果をより精度よいものにするためには以下のような改善点が挙げられる。

1. SEM 観測時に破面が歪んで観測されないようにする。
2. マッチングの精度を上げる。
3. 破壊面積計算アルゴリズムの改良。

1 に関しては破面観察時に表面に金蒸着を施して観察することで改善可能である。

2 に関しては CABIN 上で表示した際に破面の凹凸をより明瞭に表現することによってマッチングの精度を上げることができると期待できる。

3 に関しては 2 破面の衝突判定のアルゴリズムを考案することでより破壊面積を精度よく計算可能である。

また、CABIN 上で FRASTA 解析を行なう際、対応付けと破壊過程のシミュレーションは同時に行なえたが、破壊面積を CABIN 上で目に見える形で表現されないため、シミュレーションと計算を独立に行なわなければならなかった。そのため、計算後に再びシミュレーションを行ない、破壊条件変化点近傍での破壊の様子を観察しなければならず、非常に手間のかかるものであった。今後は CABIN 上でも破壊面積の推移の様子を把握できるようなシステムを考案するなどして改善していく必要がある。

参考文献

- [1] 小寺沢良一編：『フラクトグラフィとその応用』，日刊工業新聞社 (1981).
- [2] 日本電子顕微鏡学会関東支部編：『走査電子顕微鏡—基礎と応用—』，共立出版株式会社 (1976).
- [3] 大塚尚武, 小林隆夫, 渡士克己, 菊池政之：「トポグラフィ手法による高温疲労破面の三次元解析」，材料, **39**, 443, pp. 1145-1150 (1990).
- [4] T. Kobayashi and D. Shockey："FRASTA: A New Way to Analyze Fracture Surfaces", *Adv. Materials and Processes*, **140**, 5, pp. 28-34 (1991).
- [5] 小倉敬二, 三好良夫, 久保司郎, 箕島弘二：「5. 破壊力学実験における計測・モニタリング手法の進歩」，材料, **39**, 440, pp. 575-581 (1990).
- [6] T. Kobayashi and D. Shockey："Fracture Analysis Via FRASTA", *Adv. Materials and Processes*, **140**, 6, pp. 24-32 (1991).
- [7] T. Kobayashi and D. Shockey："A Fractographic Investigation of Thermal Embrittlement in Cast Duplex Stainless Steel", *Metallurgical Transactions*, **18A**, November, pp. 1941-1949 (1987).
- [8] 山口祐一郎, 野中英正：「水素割れ環境下における炭素鋼の破壊過程解析」，材料, **43**, 494, pp.1393-1399 (1994).
- [9] OpenGL ARB：『OpenGL Programming Guide (日本語版) The Official Reference Document for OpenGL, Release 1』，アジソン・ウェスレイ・パブリッシャーズ・ジャパン株式会社 (1993).
- [10] 羽原恭寛：『共役破面の三次元形状を用いたストレッチゾーン寸法の測定』，東京大学卒業論文 (1990).
- [11] 三浦憲二郎：『OpenGL 3D グラフィックス入門』，朝倉書店.
- [12] 梅基宏：『バーチャルリアリティ技術を応用した金属破断面の立体構造解析』，東京大学修士論文 (1995).
- [13] 日野幹雄：『スペクトル解析』，朝倉書店.
- [14] CABIN ホームページ
URL = <http://www.ihl.t.u-tokyo.ac.jp/Projects/CABIN/index-j.html> .
- [15] 石綿昌平：『OpenGL CABIN Library version 1.2 Users Guide』
URL=<http://www.ihl.t.u-tokyo.ac.jp/Projects/CABIN/OpenGL/usersguide/> .

- [16] 寄田知弘：『並列処理を用いた不規則荷重下の疲労亀裂進展実験システムの開発』，東京大学修士論文 (1991).
- [17] 河西朝雄，河西雄一：『VRML 入門』，技術評論社.

あとかき

謝辞

本研究は東京大学工学部 酒井信介教授のご指導のもとに進められました。研究方針に困った時、行きづまった時に先生には適切なアドバイスをして頂きました。先生の適切な助言のおかげで本研究がこのようなしっかりしたものになったのだと思います。本当にありがとうございました。

助手の高野さんにはお話の中に非常にためになるものをたくさん見出すことができました。本当にありがとうございました。

技官の浅川さんには破壊試験機に関していろいろご指導いただきました。要領の悪い私に愛想を尽かさず教えていただき感謝しております。

助手の泉さんには研究の助言は勿論のこと普段の生活にまで、色々とお世話になりました。泉さんから学ばせていただいたことがたくさんありました。私の研究がここまで進んだのも泉さんに負うところが大きかったと思います。

ドクターコースのジャンさん、ジャックサニーさんとは研究室内で楽しい会話をさせて頂きました。ありがとうございました。

修士2年の小野さん、研究生の柴田さんとは研究室で会う機会が多かったもののあまり話す機会に恵まれませんでした。話をしていると楽しかったです。

修士1年の岡本さん、研究室には来てるんですか？

修士2年の橘鷹さんと修士1年の原さんには研究室での生活面で非常にお世話になりました。お二人がいらっしゃるおかげで研究室ライフは楽しいものでした。

研究面ではIML 非常勤研究員の高垣さんと博士課程1年の山際さんには非常にお世話になりました。高垣さんには僕のCABIN実験におそくまで付き合ってもらったり、プログラムを組む上での助言をたくさんしていただきました。IMLでも非常に頼れる存在でした。本当に感謝しております。山際さんにはどんなときでもずっとお世話になりっぱなしでした。生活面でも研究面でも色々とお手伝いいただきました。特に夏には院試の心配までかけてしまって本当にすみませんでした。この1年間助けていただいたことを非常にありがたく思っています。

そして最後に卒論生の明神君と村上君にも感謝です。2人のおかげで研究を最後までがんばることができました。どうもありがとう。

他にもお世話になった全ての人に、感謝の意を込めて、このつたない文章をもって謝辞と代えさせていただきます。

付 録 A SEMの仕様

SEM(ERA-4000) の概観



図 A.1: SEM

SEM(ERA-4000) の仕様

主性能

- 1) Z 方向分解能 (凹凸測定時) $1nm$
- 2) XY 方向分解能 (SEM 観察時) $4.5nm$

主機能

- 1) 2 次電子線像機能 イメージプロセッサによるモード切替
 - (1) 差信号による凹凸 2 次電子像
 - (2) 和信号による組成 2 次電子像
 - (3) 通常の 2 次電子像
- 2) 3 次元表面解析機能 (1) X - Z, Y - Z 断面形状測定
 - (2) 3D 断面形状測定
 - (3) X - Z, Y - Z 表面粗さ解析
 - (4) 3D 表面粗さ解析

仕様性能

A. 電子光学系

- 1. 電子銃 タングステンヘヤピン
- 2. 加速電圧 $0.5 \sim 30kV$ ($0.1kV$ ステップアップ)
- 3. 二次電子像分解能 $4.5nm$
- 4. レンズ 3 段磁界レンズ
- 5. ワーキングディスタンス $8 \sim 40mm$
- 6. 倍率 $\times 10 \sim \times 300,000$
- 7. 視野移動 $\pm 30\mu m$
- 8. 焦点非点合わせ オートフォーカス&スティグ及び手動

B. 画像

- | | |
|-----------------|--|
| 1. 走査モード | FRAME, REDUCED FRAME, SPOT, LINE X, LINE Y, CB-MONITOR |
| 2. 走査速度 | TV, 0.2, 5, 20, 40, 80, 160 秒/フレーム |
| 3. 走査方向 | 360° 回転可能 |
| 4. 画像種類 | A, B, A+B, A-B, EXT |
| 5. 画像記憶 | SEM 像
576 × 440 …… 2 フレーム |
| 6. 画像寸法 | 観察用 …… 165 × 125mm
撮影用 …… 120 × 90mm |
| 7. 画像記録 (オプション) | イメージプリンタ 100 × 75mm
ポラロイドフィルム 118 × 89mm
ブローニフィルム 60 × 70mm |
| 8. スーパーインポーズ | a. ミクロンマーカー, フィルムナンバー
(オートカウント), タイマー (年月日)
b. 電源電圧値
c. コメント
d. 試料位置
e. +字線 |
| 9. 積分画像 (オプション) | 積分回数 1~255 回
積分スキンスピード Rapid(0.2 秒) |

C. 試料ステージ

- | | |
|-------------|--|
| 1. 試料寸法 | 最大径 $\phi 125 \times H 10mm$
最大厚 $\phi 50 \times H 30mm$
小試料 $\phi 10 \times H 10mm$ |
| 2. X - Y 移動 | 30 × 68mm モーター駆動 |
| 3. Z 移動 | 8~40mm |
| 4. 回転 | 360° エンドレス モーター駆動 |
| 5. 傾斜 | 0~60° (但し、試料大きさ、Z-位置により制限あり。) |
| 6. 試料交換 | 大気解放 |
| 7. ステージ送り | 最少ピッチ 5 μm |
| 8. 位置表示 | X, Y 位置 4 桁、回転 1° 単位 |
| 9. 送り速度 | 倍率連動 |

D. 排気系

- | | |
|-----------|------------------------|
| 1. 操作 | 完全自動排気 |
| 2. 試料交換時間 | 4分以内 |
| 3. 到達真空度 | 2.6mmPa |
| 4. 真空ポンプ | 油拡散ポンプ
油回転ポンプ (直結型) |

E. その他

- | | |
|---------|---------------|
| 1. 架台 | 密封式エアーマウント |
| 2. 保護機能 | 対停電、対断水、対真空劣化 |

F. 三次元表面解析機能

1) プロファイル測定

- | | |
|----------------|--|
| (1) ビーム走査 | デジタル走査 |
| (2) 測定方向 | 二次電子画像の X 方向、Y 方向切換 |
| (3) 測定位置表示 | CRT 上+字マークによる。 |
| (4) 測定の長さ | X 方向 2.4 μ m $\sim$ 60 μ m
Y 方向 1.8 μ m $\sim$ 45 μ m が適切である。
(測定長は画像倍率で設定する。) |
| (5) 測定点設定 | 最大 4096 点/1 ライン (1 ライン測定)
最大 4096 点/1 ライン (エアロビュー)
* 但し、ライン $\times$ 測定点は 250,000 以下 |
| (6) 測定ライン数設定 | 2 $\sim$ 4096 ライン (エアロビューの時設定) |
| (7) Z/X, Z/Y 比 | 0.1 $\sim$ 1000 倍 |
| (8) 傾斜補正 | 1 ライン測定時 オート・マニュアル
エアロビュー オート |
| (9) 測定データ及び記録 | |
| 1. 凹凸プロファイル | カラー CRT ディスプレイ |
| 2. ズーム機能 | 1 ライン測定の一部を拡大して
カラー CRT ディスプレイ |
| 3. エアロビュー | カラー CRT ディスプレイ |
| 4. 二点間測定 | カーソルにより 1 ライン上の相互の 2 点を
指定し、2 点の距離、高さの差、角度を
測定できる。 |
| 5. 記憶 | フロッピーディスク及びハードディスク |
| 6. CRT 表示記録 | カラーハードコピー (オプション) による。 |

2) 三次元粗さ解析

- | | |
|------------|---|
| (1) 測定方向 | X 方向及び Y 方向 |
| (2) 入力データ数 | 最大 4096 点/ライン
最大 4096 点/ライン/フレーム
(ライン × 測定点 = 250,000 以内) |
| (3) 記憶 | ハードディスク (100MB 以上) 及び
フロッピーディスク |
| (4) メモリー | 1.6MB + 8MB |
| (5) 解析機能 | 解析結果はカラー CRT にディスプレイすると
共に、カラーハードコピーが可能である。 |

解析機能

1. 等高線表示

指定されたスレッシュホールド値とステップ数により、同じ高さの場所を地図のように結ぶ。スレッシュホールド値を順次変えながらその時の切口の形状を表示することにより画像化する。

2. 切断面の面積率

指定した切断面の総面積の全測定面積に対する比率を算出、表示する。また、スレッシュホールド値と切断面積との関係をヒストグラムで示すことが出来る。

3. 指定面積あたりの山数

指定した切断面の切口の数 (山の数) を算出、表示する。(但し、切口形状が単純形状の場合に限る。) また、スレッシュホールド値と切口の数 (山の数) との関係をヒストグラムで示すことが出来る。

4. 粒度

切断面の総面積を切口の数 (山の数) で割ったもの、つまり切断面の平均面積を算出、表示する。また、スレッシュホールド値における粒度をヒストグラムで表示する。

5. 各切断面の最大高さ

各切断面の切口の最大高さ (山の高さのばらつき) を算出する。

6. 表面粗さの三次元パラメータ

表面粗さの三次元パラメータ R_a , R_z , R_{max} を算出する。

- R_a …… 中心線平均粗さ
- R_z …… 10 点平均粗さ
- R_{max} …… 最大高さ

R_a , R_z については、平均値、最大値、最小値を算出する。また、グラフにすることも出来る。

7. 表面積代替値

測定された形状データから表面積を計算する。また、二次元平面 (測定面) の面積との比をとることも出来る。(表面積率)

8. データのフィルタリング機能

測定された形状データに対し指定した周波数による高域、低域、中域フィルタリングをかけることが出来、結果のデータは1～8の解析にかけることが出来る。

9. 鳥瞰図表示

鳥瞰図表示を行なうことが出来る。Z方向倍率、視野(角度)は入力により設定することができる。

10. BATCH 処理

解析を1個ずつ実行するのではなくて、処理順序を決めたFILEを作成し、そのFILEに従って順次解析を連続的に行なっていく機能である。無人で長時間の処理をいくつか行なう時に便利である。(ただし、解析データは全てハードディスク上にFILEされている必要がある。)

処理順序はユーザーが自由に作成、変更、削除することができる。

装置寸法

1. 本体	$W540 \times D750 \times H1350mm$	265kg
2. 操作	$W1000 \times D750 \times H1200mm$	90kg
3. 油回転ポンプ	$W500 \times D260 \times H410mm$	30kg

付 録B CABINのスペック

CG Rendering

CPU	MIPS R4400
FPU	MIPS R4000
CLOCK	250MHz
Main Memory	256MB
Texture Memory	16MB
Rendering Engine	Infinite Reality (10M Polygon/s)

表 B.1: CG Rendering

共有メモリ

SCRAMNet (TM)	容量 1Mbyte
---------------	-----------

表 B.2: 共有メモリ

立体視

通常モード	1280pixelx1024pixel 60Hz スクリーン表示部分 中央正方形部分 1000x1000
立体視モード	1024pixelx768pixel 96Hz スクリーン表示部分 中央正方形部分 750x750

表 B.3: 立体視

トラッキング

Ultratrak	磁場を生成するソース部と、磁気をセンシングするセンサ部から成り立つ 取り付け可能なセンサ数 16 個（これを station と呼ぶ） 実空間三次元の 6 自由度（位置、角度）のデータを返す 分解能 2 インチ UDP にて bottom と接続
-----------	--

表 B.4: トラッキング

同期

RGB 信号同期	Genlock を用いている 特に立体視をする場合に必要 Front がソース
共有メモリアクセスセマフォ	開発者がソフトウェア的に実装

表 B.5: 同期

入力デバイス

Nintendo 64 コントローラ	信号変換機を通して, RS232C 経由で Front の tty ポート番に接続
振動ポインタ	信号変換機を通して, RS232C 経由で Front の tty ポート番に接続

表 B.6: 入力デバイス

付 録 C 開発プログラム

プログラムの流れ

CABIN での OpenGL プログラムは図 C.1 のような流れをとる.

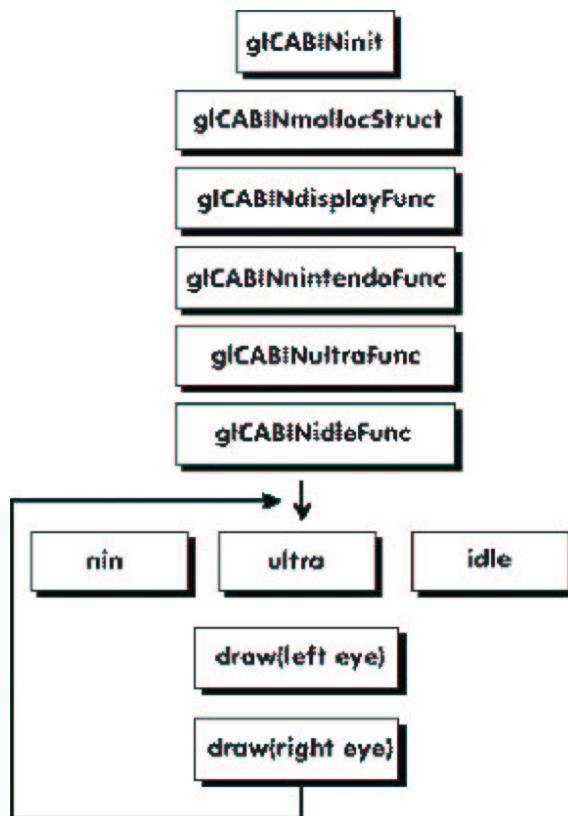


図 C.1: プログラムフロー

`glCABiNinit` において, `genlock` の設定, デバイスのメモリ領域の確保, Xwindow 周りの設定を行う.

`glCABiNmallocStruct` でユーザの指定する構造体を共有メモリ内に確保する.

`glCABiN***Func` において, デバイス制御関数, 描画関数の指定を行う.

`ultra`, `nintendo`, `idle` などにおいて実際に共有メモリからデータの取得, 制御を行う.

`draw` において描画を行う. プロジェクション行列の決定, モデルの決定など. 両眼立体視の場合は, 左目, 右目のそれぞれを描画するので2回呼ばれる. 特に, `glCABiNprojectino()` をもちいれば, 5面すべてに CABIN 特有のプロジェクション行列が設定される.

main関数

本研究で開発したプログラムの main 関数は以下のである.

```
void
main(int argc, char **argv)
{
    int i;
    double st, en;

    m = 50.0;
    n = 50.0;

    strcpy(semdata[0].filename, argv[1]);
    strcpy(semdata[1].filename, argv[2]);

    ZX = 1;
    for(i = 3; i < argc; i++){
        if(strcmp(argv[i], "-z") == 0) {
            ZX = atof(argv[i+1]);
            break;
        }
    }

    glCABINinit(argc, argv); /* initialize for CABIN */
    fprintf(stderr, "host = %d\n", GLCABINindex);

    myshare = (MYSHARE *)glCABINmallocStruct();
                /* get the Structure on Scramnet */

    initialize(); /* initialize for user */

    readdata();
    colorSet();
    glCABINdisplayFunc(display);
    glCABINidleFunc(idle);
    glCABINnintendoFunc(nintendo);
    glCABINultraFunc(ultra);
```

```

    glCABINmainLoop(); /* main loop */
}

```

起動方法は実行コマンドの後ろに表示する破面のデータファイルを下, 上の順に指定する. この2つのファイルは strcpy コマンドで読み取りそのファイルを参照する. readdata() 関数でファイルのデータを読み取り, z 座標の大きさによって各頂点の色を指定する関数が colorSet() である.

glCABINdisplayFunc(display) において関数 display() を呼び出す. display() 関数については次節で説明する.

glCABINidleFunc(idle), glCABINnintendoFunc(nintendo), glCABINultraFunc(ultra) は CABIN に特有な関数であり, glCABINidleFunc(idle) はアイドル状態用関数の指定を行い, glCABINnintendoFunc(nintendo) では Nintendo64 コントローラ (入力用デバイス) 用関数の指定, glCABINultraFunc(ultra) では Ultratrak 用関数の指定を行なう.

glCABINmainLoop() がメインループで glCABINanimate を呼び出す.

display 関数

display() 関数は以下の通りである.

```

void
display(void)
{
    short i,j;
    float matdiff[4] = {1.0, 1.0, 1.0, 1.0};
    GLfloat material_color0[4] = {1.0, 0.0, 0.0, 1.0};/*lightdiffuse*/

    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glMaterialfv(GL_FRONT, GL_DIFFUSE, matdiff);
    drawphase();/*draw phase*/

    for (i = 0; i < 2; ++i)
    {
        for (j = 0; j < 3; ++j)
        {
            printf("move[%d].face[%d].x = %lf\n", i, j, move[i].face[j].x);
            printf("move[%d].face[%d].y = %lf\n", i, j, move[i].face[j].y);
        }
    }
}

```

```

    printf("move[%d].face[%d].z = %lf\n", i, j, move[i].face[j].z);
}
printf("flgm = %d flgs = %d\n", flgmove, flgsurface);
}
printf("calc0\n");
calc();
printf("frac = %d\n", semdata[0].ScanNum * semdata[0].DataNum - v);
v = 0;
printf("s = %lf\n", s);
s = 0.0;
bothinit();
}

float matdiff[4] , GLfloat material_color0[4] ,
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT),
glMaterialfv(GL_FRONT, GL_DIFFUSE, matdiff)

```

に関しては専門書 [9, 11] を参考されたい.

`drawphase()` は破面の高さデータから破面を表示する関数で、中に移動、回転のアルゴリズムも含まれている. `main` 関数の中の n は表示する際にそのままの大きさでは破面が画面に入りきらないので破面全体を $1/n$ 倍して表示している. m は上破面を表示する際最初に m だけ高さ方向に移動させて表示させるために用いられている. 各破面の初期位置からの移動距離は `move[].face[].x`, `move[].face[].y`, `move[].face[].z` によって管理されている. ここで `move`, `face` とともに構造体である. 変数 `flgm` と `flgs` は移動・回転のためのフラグで, `flgm` は移動あるいは回転を指し, `flgs` は対象となる破面を指す. v は下破面が対応する上破面よりも高い位置にある点の数を表し, s は破壊面積を表す変数である. この2つの値は計算の関数の中でそれぞれ $v++$, $s++$ で表されているので常に初期化しなければならない. `bothinit` では両破面が共に移動するとき移動距離を初期化しなければならないので, そのための関数である.

以上

1～ 71 ページ 完

卒業論文

平成13年 2月 9日 提出

90262 山田 直秀