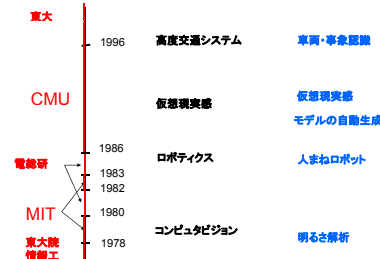


時空間情報技術を支える 画像処理技術

東京大学
先進モビリティ研究センター
(大学院 情報学環)
池内克史

自己紹介

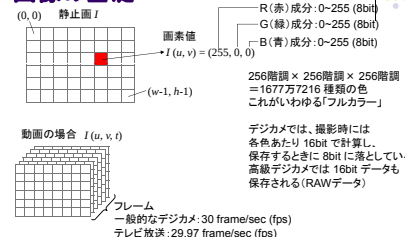


所属

- 大学院
 - 学際情報学府・先端表現情報学コース
 - 情報理工学系研究科・コンピュータ科学専攻
 - 情報理工学系研究科・電子情報学専攻
- 研究所
 - 生産技術研究所
 - 空間情報科学研究センター

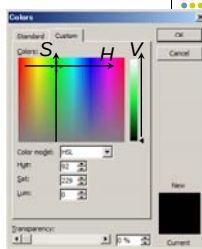
画像処理の基礎

画像の基礎



表色系(色空間)

- 色は3パラメータ
- (直交座標と極座標があるのと同じように) RGB 以外の表し方もある
- HSV(HSL)
 - H = Hue (色相)
 - S = Saturation (彩度)
 - V = Value (明度)
- YCbCr テレビ映像用
- CIELab 人間知覚に線形



画像の「形式」

- 静止画
 - 圧縮なし: bmp, ppm, pgm
 - 可逆圧縮: png, gif
 - 不可逆圧縮: jpg
- 動画
 - 動画形式 (avi, mpg, wmv, mov, ...)

デジタル画像処理(2D)

デジタル画像とは

- アナログ情報 (フィルム、絵、実世界)

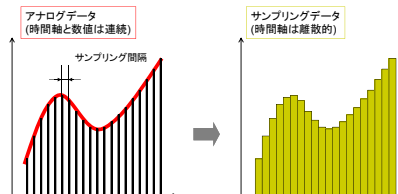
量子化と標本化

- デジタル画像
 - デジタルカメラ
 - 携帯電話
 - PCデータ, IT
 - デジタル放送



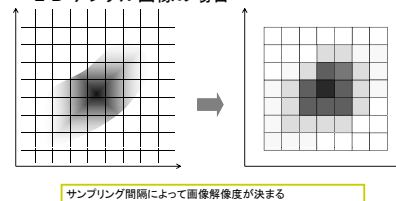
標本化

- アナログデータを離散的に領域分割



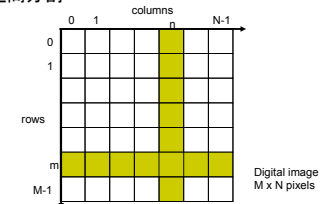
標本化(続き)

- 2-D デジタル画像の場合



ピクセル (Pixel)

- 2-Dデジタル画像の単位
- 空間分割

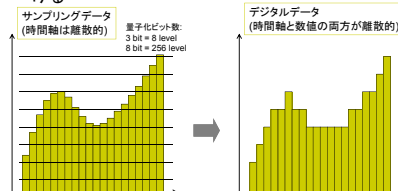


空間的標本化(解像度)



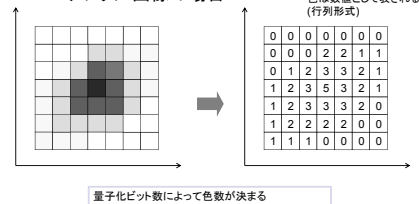
量子化

- サンプリングされたデータの数値を離散的に分ける



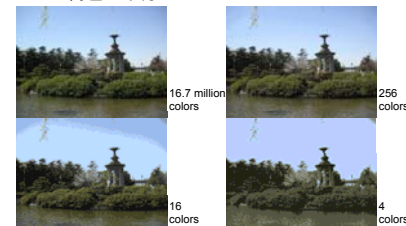
量子化(続き)

- 2-Dデジタル画像の場合



表現の色数

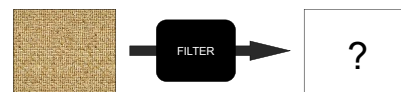
- 何色で十分か?



フィルタ処理

フィルタ

- 人間の視覚やコンピュータビジョンための前処理
 - ノイズ除去
 - 画像強調
 - 特徴抽出

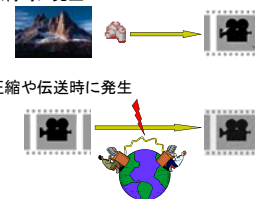


空間フィルタ・周波数フィルタ

- 空間領域での処理
 - 隣接ピクセルとの演算
- 周波数領域での処理
 - 画像をフーリエ変換

ノイズ除去

- ノイズ源
 - 取得時に発生
- 圧縮や伝送時に発生



平均値フィルタ

- 隣接点の値を平均して置き換える

0	5	4	6	4
3	7/9	8/9	5/9	5
5	9/9	10/9	8/9	8
3	7/9	8/9	0/9	9
1	5	7	9	10

3 x 3
(5 x 5)
(7 x 7)

0	5	4	6	4
3	7	8	5	5
5	9	7	8	8
3	7	8	8	9
1	5	7	9	10

平均値フィルタ(続き)

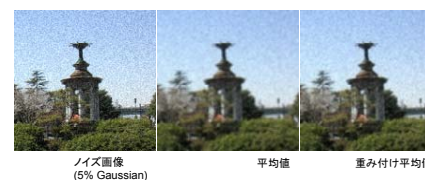
- 重み付け平均では

0	5	4	6	4
3	7/16	16/16	5/16	5
5	18/16	40/16	16/16	8
3	7/16	16/16	0/16	9
1	5	7	9	10

0	5	4	6	4
3	7	8	5	5
5	9	8	8	8
3	7	8	6	9
1	5	7	9	10

平均値フィルタ(続き)

- 例1 ガウスノイズの場合



平均値フィルタ(続き)

- 例2 ショットノイズの場合



非線形フィルタ

- 最大値フィルタ
 - ウィンドウ内の最大値で置き換える
- 最小値フィルタ
 - ウィンドウ内の最小値で置き換える
- メディアン(中央値)フィルタ

メディアンフィルタ

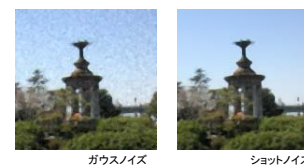
- ウィンドウ内の中央値で置き換える

0	5	4	6	4
3	7	8	5	5
5	9	10	8	8
3	7	8	0	9
1	5	7	9	10

0	5	4	6	4
3	7	8	5	5
5	9	8	8	8
3	7	8	8	9
1	5	7	9	10

メディアンフィルタ(続き)

- 例 3 x 3 フィルタ



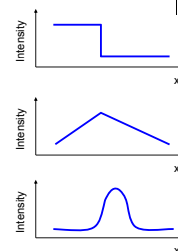
エッジ

エッジ

- 画像中の明るさ(数値)が急激に変化する部分
- 近傍ピクセルとの微分処理(離散的には差分式)
- 物体認識などでは重要な特徴となる
- 微分処理であるためノイズに弱い

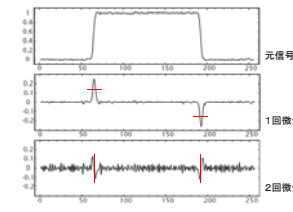
エッジの種類

- ステップエッジ
- ルーフエッジ
- ピークエッジ



1-D エッジの微分

- 1回微分と2回微分



Gradient-base

- 1回微分のオペレータ

$$\nabla f(x, y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

- 離散的な差分式では

$$f_x(m, n) = f(m+1, n) - f(m, n)$$

$$f_y(m, n) = f(m, n+1) - f(m, n)$$

$$f_x(m, n) = f(m+1, n) - f(m-1, n)$$

$$f_y(m, n) = f(m, n+1) - f(m, n-1)$$

エッジの強さと方向が得られる

Gradient-base (続き)

- オペレータの種類

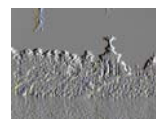
- Roberts $D_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ $D_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

- Prewitt $D_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$ $D_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$

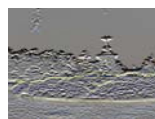
- Sobel $D_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$ $D_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

Gradient-base (続き)

- 例 Prewittオペレータ



D_x



D_y

Laplacian operator

- 2回微分のオペレータ

$$\nabla^2 = D_x^2 + D_y^2 = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} + \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \begin{matrix} (D_x^2 = D_x \cdot D_x) \\ (D_y^2 = D_y \cdot D_y) \end{matrix}$$

$$\nabla^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \nabla^2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

4 direction 8 direction

エッジの強さが得られる

Laplacian operator (続き)

- 例 Laplacianオペレータ



4 direction



8 direction

Laplacian Of Gaussian

- 微分演算はノイズに対して弱い
- Gauss関数でぼかして(ノイズ除去)してからLaplacianオペレータ

$$G(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

- Laplacian of Gaussian

$$\nabla^2 G(x, y) = \frac{-1}{2\pi\sigma^4} \left(2 - \frac{x^2+y^2}{\sigma^2} \right) e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Laplacian Of Gaussian (続き)

- 例 LOGオペレータ



$\sigma=1$



$\sigma=2$

三次元画像処理

コンピュータビジョン

- 人間の視覚機能の代替アルゴリズムを設計する分野
- 2次元画像(網膜画像)から3次元世界を再構成
- 最大の応用分野はロボットが行動するための目

2次元画像と3次元世界

- カメラ画像は、三次元の座標系から二次元の画像座標系へ射影したもの

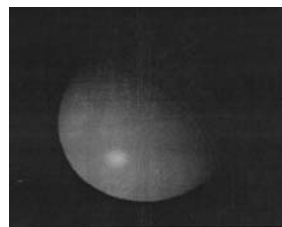


二次元画像

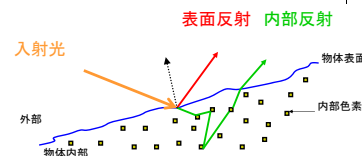


三次元世界

なぜ球に見えるのか？

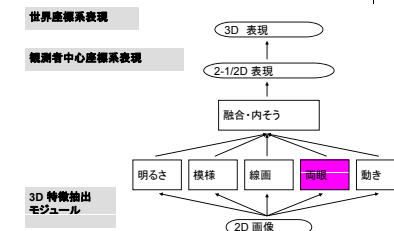


表面反射と内部反射

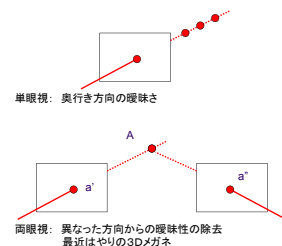


表面反射 = ハイライト、光源色、偏光、高い方向性
内部反射 = 物体色、非偏光、全ての方向へ
プラスチックや塗装面は両方の反射があり
金属面は表面反射のみ

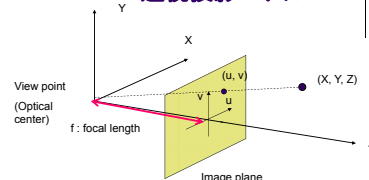
Marr のパラダイム



両眼立体視

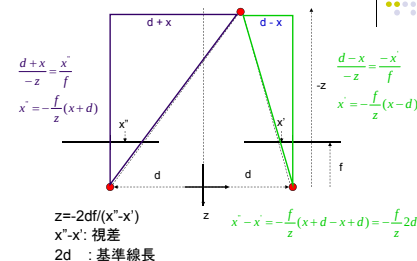


透視投影モデル



$$\frac{u}{f} = \frac{X}{-Z} \quad (u, v) = \left(f \frac{X}{-Z}, f \frac{Y}{-Z} \right)$$

基本方程式



基本原理 特徴と戦略

対応づけのための特徴

a. 明るさ分布



b. エッジ



c. エッジ間距離



d. インタレスト点

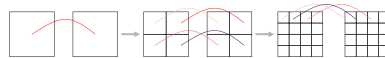


対応付けのための戦略

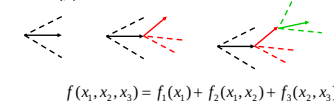
a. 弛緩法



b. 粗密法

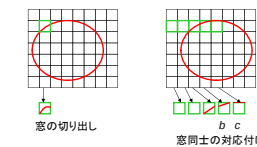


c. DP法

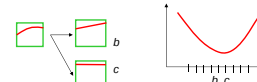


領域法両眼立体視

ウィンドマッチング



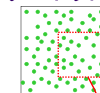
問題点
窓のサイズ: 小さい: 精度向上、大きい: 安定
平坦な最小値



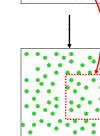
両眼立体視の実例

領域法

ランダムドットの作り方



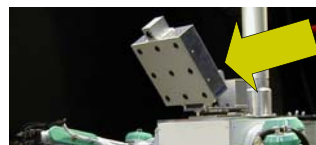
- 左画像の一部を切り出し



- シフトして右画像に埋め込む

- 人間には板が見える

領域法ハードウェア



小松製 9眼ステレオ

ハードウェア化しやすい
GPUですぐ出来る

領域ステレオ処理結果

◆ 入力画像列



◆ 出力(距離画像)



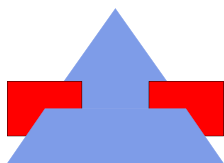
- 窓内明るさ照合
- エッジボラ拘束
- 視差限界
- 明るい点: 近い、暗い点: 遠い
- 窓処理のため周辺で拡大
- 視差制限のため背景がノイズ

Marr, Poggio "Cooperative computation of stereo disparity" Science 194, 283-287

MARR-POGGIO STEREO: 人間の視覚機能のシミュレーション

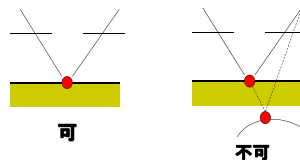
拘束(その1)

- エッジボラ拘束:
2次元探索 → 1次元探索



拘束(その2)

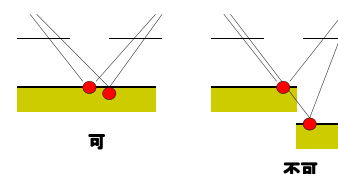
- 唯一性拘束(Uniqueness Constraint):
画像中の1点は1つだけの距離値を持つ



例: 水面と水中の石

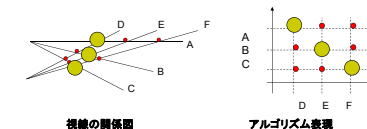
拘束(その3)

- 連続性拘束(Continuity constraint):
隣あった点は同じような距離値を持つ



アルゴリズムの定式化

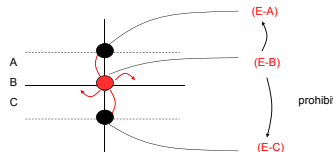
- エッジボラ拘束: 2D → 1D



各ノードを仮想的な神経細胞と考える
各ノードの興奮状態で面の存在を記述

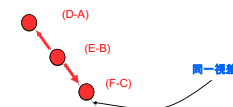
唯一性拘束のアルゴリズム化

- 同一の視線上で1つ以上のノードの興奮を禁止
- 同一視線(水平線と垂直線)にそって興奮細胞が抑制信号を送る



連続性拘束のアルゴリズム化

- 同一視差での興奮を励起
- 同一視差(斜め線)にそって興奮細胞が興奮信号を送る



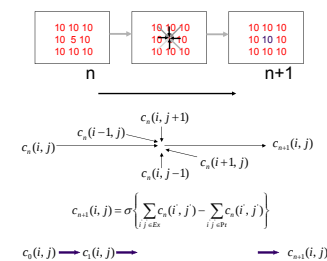
全体の定式化

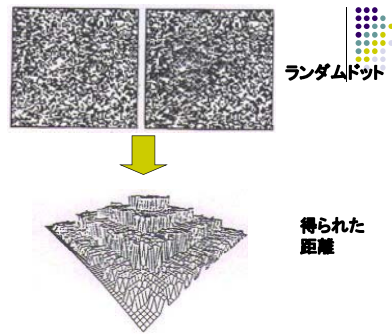
閾値関数

$$c_{n+1}(i, j) = \sigma \left\{ \sum_{i'j' \in Ex} c_n(i', j') - \sum_{i'j' \in Pr} c_n(i', j') \right\}$$

斜めよりの興奮信号 水平・垂直よりの抑制信号

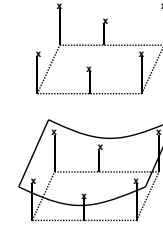
弛緩法



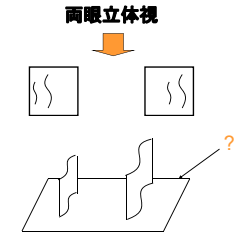


内そう

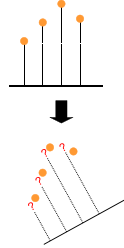
内そうとは？



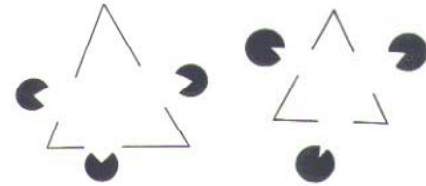
目的 1



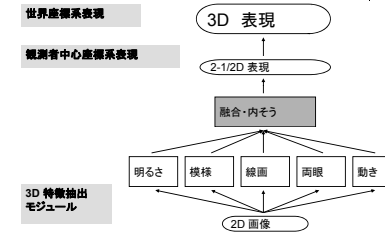
目的 2 (メッシュ化)



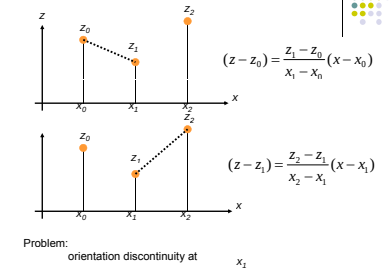
目的3: 心理学的実験の説明



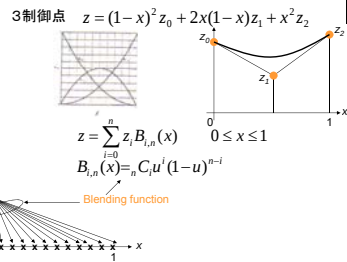
Marrのパラダイム



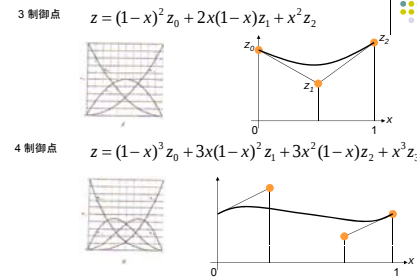
単純な方法



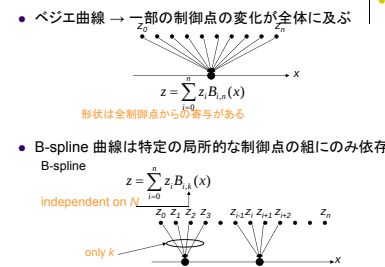
ベジエ曲線



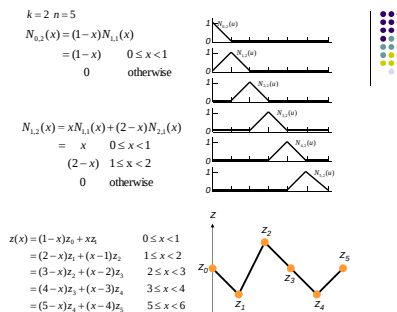
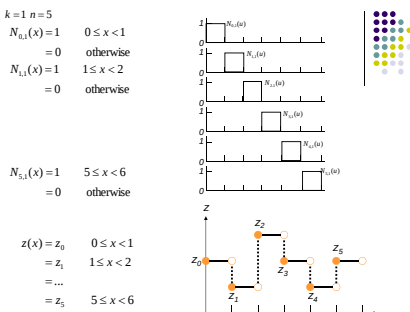
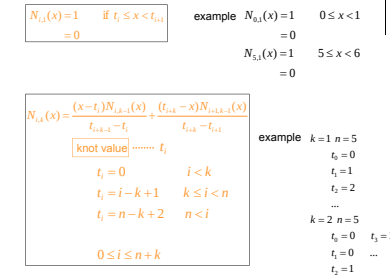
ベジエ曲線の例



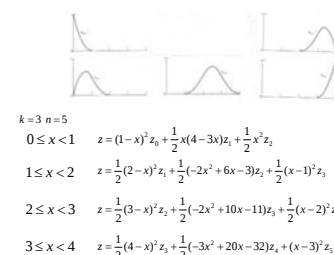
B-spline 曲線



blending 関数の一般形



B-spline 曲線の例



スネーク(エネルギー最小曲線)

- 以下のエネルギーの和を最小化

1) 内部エネルギー — まげ

$$E_{\text{internal}} = \int_0^S \text{bending-energy}(s) ds$$

2) 外部エネルギー — たとえば明るさ

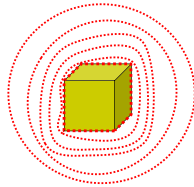
$$E_{\text{external}} = \int_0^S I(s) ds$$

3) 拘束エネルギー — 制御点

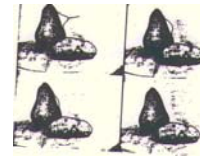
$$\text{energy stored at this spring } E_{\text{constraint}} = k(z(s) - z_0)^2$$

スネークの例

- 形状をじょじょに変形させ目的とする形状を表現

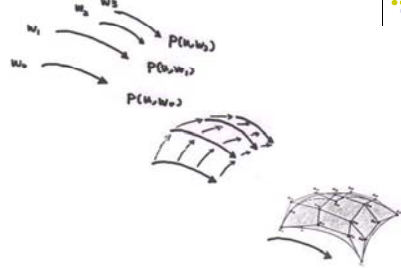


スネークの例



SNAKES:
Active Contour
Models

ベジエ と B-spline 曲面



ベジエ曲面

$$P(u, w) = \sum_{i=0}^m \sum_{j=0}^n P_{i,j} B_{i,m}(u) B_{j,n}(w) \quad 0 \leq u \leq 1, 0 \leq w \leq 1$$

vertices of the characteristic polyhedron

for $m=3 \quad n=3$

$$= \begin{bmatrix} (1-u)^3 3u(1-u)^2 3u^2(1-u)u^3 \end{bmatrix} \times P \times \begin{bmatrix} (1-w)^3 \\ 3w(1-w)^2 \\ 3w^2(1-w) \\ w^3 \end{bmatrix}$$

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{bmatrix}$$

B-spline 曲面

$$P(u, w) = \sum_{i=0}^m \sum_{j=0}^n P_{i,j} N_{i,k}(u) N_{j,k}(w)$$

independent on m, n

The shape approximates the characteristic polyhedron. The approximation is weaker the higher the value of k . $N_{i,k}, N_{j,k}$ are calculated as for 2D curves.

$$P_{s,t}(u, w) = U_k M_k P_{k,l} M_l^T W_l^T$$

$$U_k = \begin{bmatrix} u^{k-1} & u^{k-2} & \dots & u & 1 \end{bmatrix}$$

$$W_l = \begin{bmatrix} w^{l-1} & w^{l-2} & \dots & w & 1 \end{bmatrix}$$

$$P_{k,l} = P_{l,j} \quad \begin{matrix} s-1 \leq i \leq s+k-2 \\ t-1 \leq j \leq t+l-2 \end{matrix}$$

M_k, M_l are same to 2D case

デフォーマブル曲面(エネルギー最小曲面)

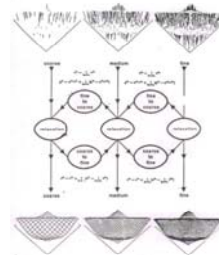
$$\varepsilon(v) = \iint (\text{deformed-energy(internal)} dx dy + \iint \text{external-energy} dx dy + \int \frac{1}{2} k (z(x_i, y_i) - c(x_i, y_i))^2$$

find z which minimize ε (calculus of variations)

$$\frac{\partial \varepsilon}{\partial u_{i,j}} = \frac{1}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2) - \frac{2}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2) + \frac{1}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2) + \frac{2}{h^2} (u_{i,j}^2 - u_{i,j}^2 + u_{i,j}^2) - \frac{2}{h^2} (u_{i,j}^2 - u_{i,j}^2 + u_{i,j}^2) + \frac{2}{h^2} (u_{i,j}^2 - u_{i,j}^2 + u_{i,j}^2) + \frac{1}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2) - \frac{2}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2) + \frac{1}{h^2} (u_{i,j}^2 - 2u_{i,j}^2 + u_{i,j}^2)$$

$$\frac{\partial \varepsilon}{\partial u_{i,j}} = 0 = 20u_{i,j} - 8(u_{i,j} + u_{i,j} + u_{i,j} + u_{i,j}) + 2(u_{i,j} + u_{i,j} + u_{i,j} + u_{i,j}) + (u_{i,j} + u_{i,j} + u_{i,j} + u_{i,j})$$

$$(i+1 \text{th}) \leftarrow \frac{1}{h^2} \times i \text{th}$$



random dot stereo pair

fine

medium

coarse

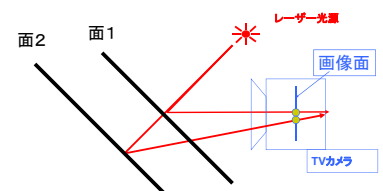
disparity from MPG stereo interpolation



disparity

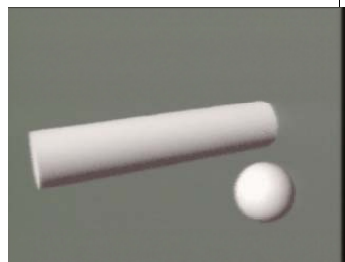
depth map

三角測量型の原理



光る位置の違いから奥行き情報が得られる

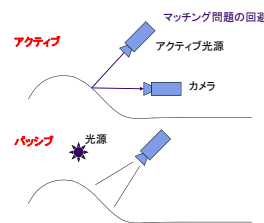
距離センサー



アクティブとパッシブ

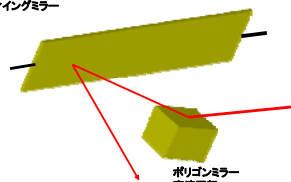
アイデア:

レーザーマーカーをシーンに投影



レーザー光線の方向変化

ノディンギミラー

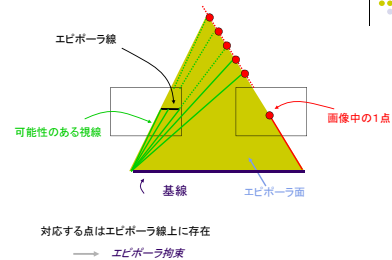


ポリゴンミラー
高速回転

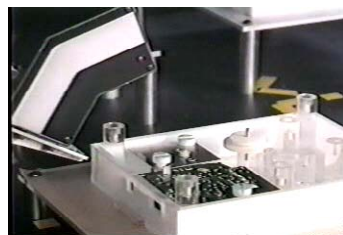
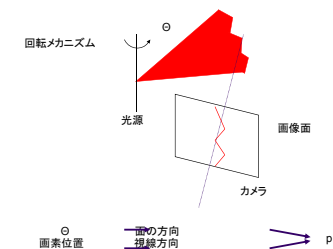
例



エビポーラ拘束



面光源投影法



Vivid 910 / 9i

- コニカミノルタセンシング(日)
- 700万円程度

- 仕様・特徴
 - 精度: ± 0.01 mm
- 距離計測原理
 - 三角測量方式

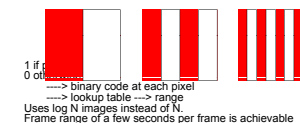


三角測量の問題点

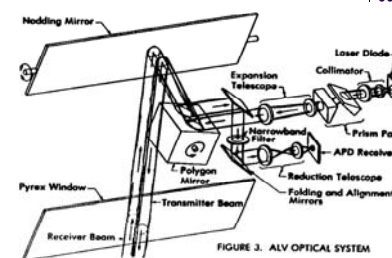
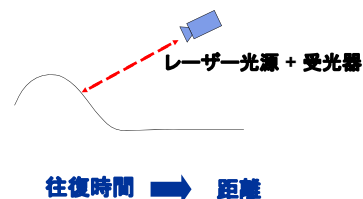
- slow:
 - $256 \times 1/30 = 8.5$ sec for 256×256 range image (+ processing time)
- calibration: source and camera must be carefully calibrated
- shadows: source and camera are not aligned
 - > shadows in range image

フレームレートの改良法?

multiple patterns:
grids, multiple lines.....
----> back to the matching problem!!!
coded patterns:



距離センサー(往復時間型)

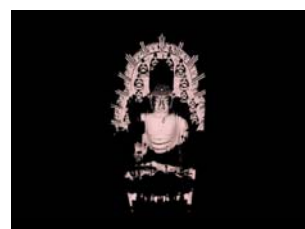


CYRAX センサー

- ◆ 往復時間方式
- ◆ 範囲: 3m ~ 100m
- ◆ 精度: 5mm

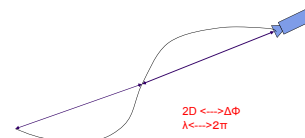


CYRAセンサーによる距離画像



変調方式距離センサー

$$D = (N/4\pi) \Delta\phi$$

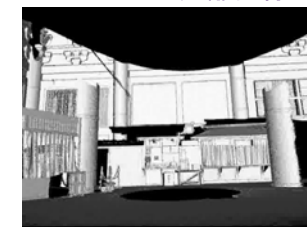


ZFセンサー

- AM変調方式
- 範囲: 0.6m ~ 50m
- 精度: 5mm
- 全周型



ZFセンサーによる距離画像



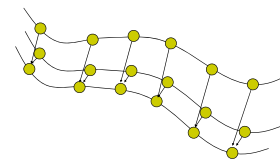
(1回のスキャン)

位置あわせ

多数の距離画像
→ 相対関係の復元が必要



位置合わせ



重ね合わせ → 相対関係

位置合わせアルゴリズム

- 特徴:
 - 点と点
 - 点と面
 - 特徴点と特徴点
- 評価関数
 - L-1
 - L-2
 - ロバスト関数

距離関数

$$\min_{R, t} \sum_{i,j,k} (R_M \vec{n}_i \cdot \{(R_S \vec{y}_j + \vec{t}_S) - (R_M \vec{x}_k + \vec{t}_M)\})^2$$

高速位置合わせ

(大石)

06)

- 線形化手法の導入
 - グラフィクスハード利用の対応点探索
- ↓
- ◆ 既存手法にくらべて1000倍の高速化

線形化

$$\varepsilon = R_m \vec{n} \cdot ((R_s \vec{y} + \vec{t}_s) - (R_m \vec{x} + \vec{t}_m))$$

$$R \approx \begin{pmatrix} 1 & -C_1 & C_2 \\ C_1 & 1 & -C_3 \\ -C_2 & C_3 & 1 \end{pmatrix}$$

線形化

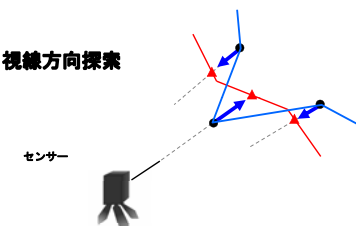
$$\varepsilon^2 = \min_{\vec{s}} \sum_{i,j,k} \|A_{ijk} \vec{s} - s_{ijk}\|^2$$

全ての距離画像間について

$$\vec{s} = \vec{n} \cdot (\vec{x} - \vec{y})$$

高速アルゴリズム基本アイデア

視線方向探索



グラフィクスハードの使用

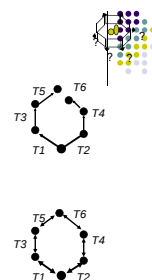
- 対象画像の各メッシュに固有のインデックス値
- インデックス値を固有の色に変換、メッシュ描画
- 色による探索

高速アルゴリズム → 100倍以上の高速

大規模位置合わせ

- 逐次位置合わせ
→ 誤差の集積

- 同時位置合わせ



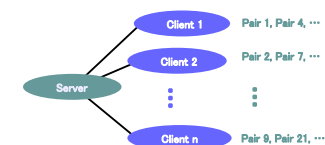
段階的データのグループ化

- 大域的な接続関係
- 局所的重なり領域の大きさ
- 相対画像間距離

並列同時位置合わせ

移動行列の計算 → サーバ

対応点探索, 誤差行列の計算 → クライアント



並列同時位置合わせ

- 線形化手法の導入
- バウンディングボックス利用の対応点探索
- PCクラスター利用並列処理

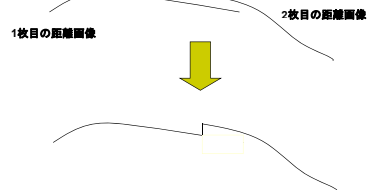


- 既存手法では扱えない大量データ処理



統合

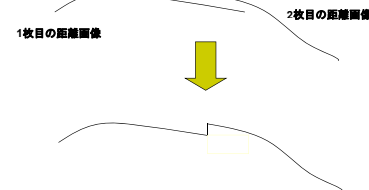
統合



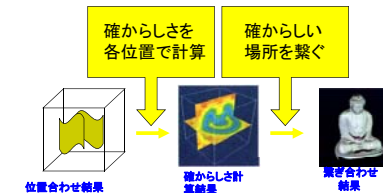
統合

- Zipper type
- Volume type

貼り合わせタイプ

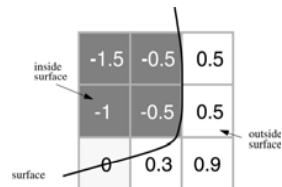


体積計算タイプ



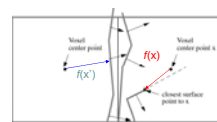
陰関数表現

- 各グリッドにおける陰関数の値を計算
- ゼロ面の生成



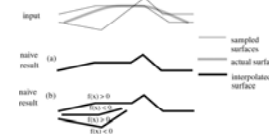
陰関数の計算

- (以前の) ナイブなアルゴリズム
 - 関数の値: 最短データ面までの距離
 - 関数のサイン: 内側か外側か



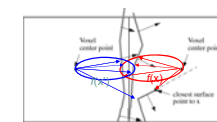
ノイズの影響

- 面ががたがたする
- 符号の不安定さ: 内一外一内のトランジット

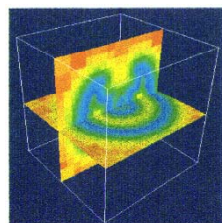


コンセンサス面

- 法線と位置によるデータのグループ化
- グループ化されたデータの平均距離

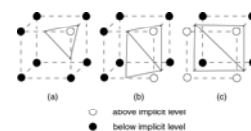


陰関数の例



ゼロ面の形成

- マーチングキューブ法: Lorensen and Cline ('87)



並列化

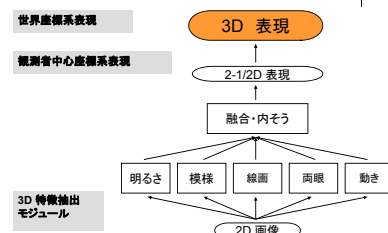
- PCクラスターによるデータ分散
- PCクラスターによる処理の並列化

PCクラスター上での統合例

- 計算時間: 23 min
 - 8 PC (dual PentiumIII 800MHz, 1GB memory)
- 130万点, 550万 ポリゴン



Marr のパラダイム

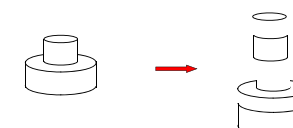


表現法の分類

- 表面表現法
 - 基本面の集まりで表現
- 関数表現法
 - 関数形(生成ルール)で表現
- 体積表現法
 - 基本体積の集まりで表現

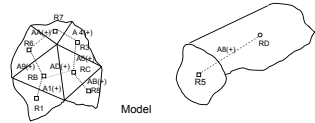
表面表現法

- 基本面の集合として表現

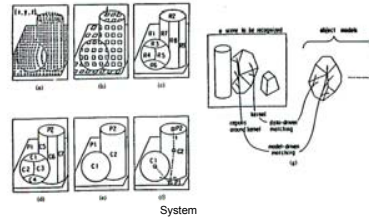


大島白井システム

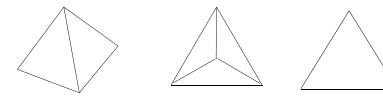
- 見えに基づく面表現
- 法線方向の不連続



モデルの使用法

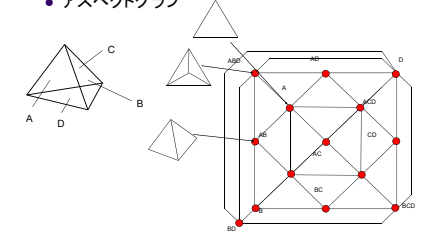


いくつかの見えが必要か？



アスペクト表現

- アスペクト = トポロジ的に見えを分類
- アスペクトグラフ



アスペクトの生成法

Face Label

F1 F2 F3 F4 F5 F6 F7 F8
1 1 1 0 0 0 0 0

Visible Invisible



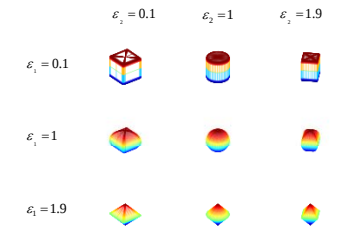
アスペクト法を用いた認識システム



関数ベース表現

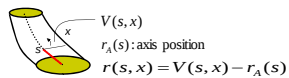
- 超楕円面
- 一般化円筒
- 拡張スネーク
- 球面関数

超楕円面



Symmetry seeking 表現

Snakesの3次元版



エネルギーの定義

内部エネルギー

軸の位置をなるべく保つ

$$E_{axis} = \int a[V(s) - V_A(s)]^2 ds$$

なるべく円形の形状

$$E_{cir} = \iint b[r(s) - |r(s, x)|]^2 ds dx$$

外部エネルギー

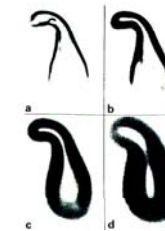
明るさの勾配のある位置に形状

$$E_{grad} = \iint -c[G \cdot \nabla^2 I(V(s, x))]^2 ds dx$$

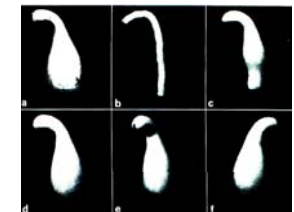
内部エネルギー+外部エネルギーを最小化

画像からのポテンシャル場

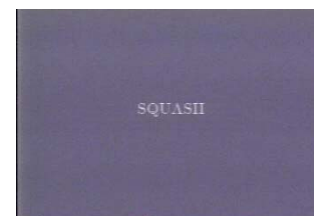
外部エネルギー場



形状生成過程

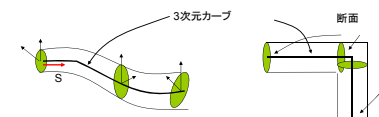


形状生成過程(ビデオ)

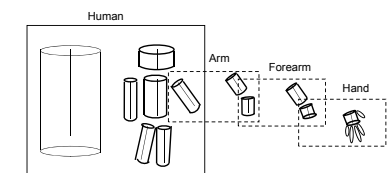


一般化円筒

空間中の軸曲線とそれにそった断面形状の変化による体積

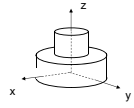


モジュール表現

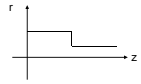


体積表現法

物体固有の座標系で表現



$$r(z) \geq x^2 + y^2$$



1. 蓄積が容易

2. 使用が難しい



2次元・3次元画像処理

- 2次元表現
- 3次元
 - 両眼立体視
 - 距離センサー
 - 内そう
 - 物体表現

