

画像情報処理

カメラの幾何学

今回の内容

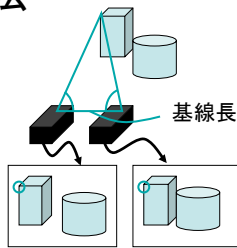
- カメラの幾何学
 - 3次元空間と2次元画像の関係
 - カメラの数式表現
 - カメラの各パラメータの推定 (キャリブレーション)

ステレオ法

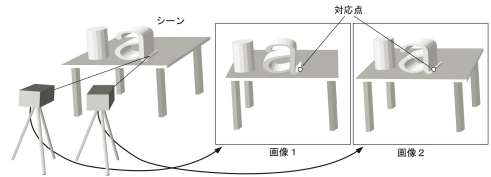
ステレオ計測

- 二つ以上の視点を使用 (人間の両目に相当)
- 左右の画像の「ずれ」を利用 (三角測量法)
- 太陽光・室内照明などが必要
- テクスチャのない平坦な部分の距離計測が困難
例: 真っ白な壁や滑らかで曲面的な物体
- 実質的な空間分解能が低い
- 対応点探索の計算量が大きい・安定度が低い

→ハードウェアによる高速化・多眼による安定化



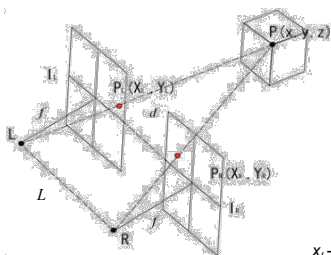
ステレオ法



- 左右の画像間で対応点を探す必要あり
 - 平坦な部分では形状計測不可
 - 対応点探索の計算量が大きい

ステレオマッチング

左右画像の特徴点位置から以下の関係が成り立つ



$$x = \frac{x_L + x_R}{2} \frac{L}{x_L - x_R}$$

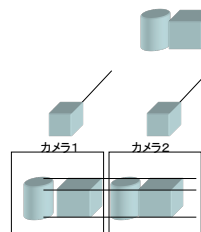
$$y = y_L \frac{L}{x_L - x_R}$$

$$z = f \frac{L}{x_L - x_R}$$

$x_L - x_R$ は両眼視差を表している (距離は視差に反比例する)

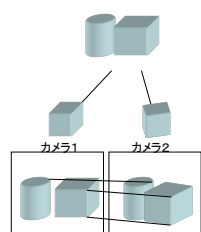
カメラの配置と画像処理

光軸が並行な場合



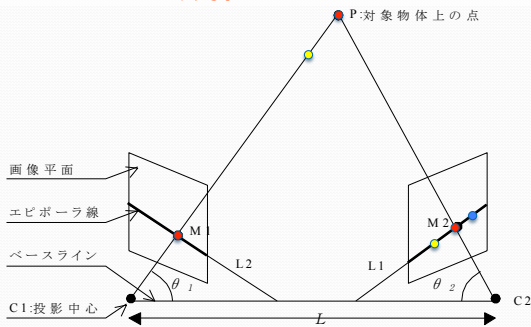
×視野の共通領域が狭い
○エピポーラ線が平行

光軸が並行でない場合



×エピポーラ線が放射状
→画像処理で解決可

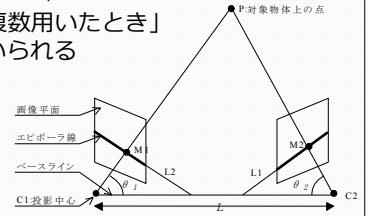
エピポーラ幾何



2台のカメラが固定されているとき、点pがそれぞれの画像のどこに写るのかを決める法則。例えば点Pが、カメラC1の画像上で座標M1に写ることがわかれば、その点PはカメラC2の画像上ではある直線L1上に写る。画像全体のどこにでも現れるわけではない。

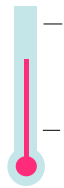
エピポーラ幾何のメリット

- ステレオカメラでの距離計測
 - 一方のカメラ画像に写っている物体上の1点か、もう一方のカメラ画像のどこに写っているのか？
 - エピポーラ幾何を使わない場合：画像全体を縦・横の2次元的に探す必要がある。時間がかかる。
 - エピポーラ幾何を使った場合：ある1本の直線上だけを探せば良い。時間が短縮され、精度も向上する。
- 他にも、「カメラを複数用いたとき」の画像処理で広く用いられる

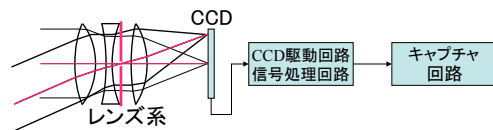


キャリブレーション

- 機器の構造からのキャリブレーションは困難
 - 実焦点距離(フォーカシング距離により可変)
 - 投影中心(レンズの中、計測困難)
- 大きさ、位置が既知の物体を計測してキャリブレーション
 - cf. 温度計の較正(氷水、沸騰水)
 - 熱膨張係数の二次成分が0と仮定
 - 画像の場合も同様、レンズ歪みなどをモデリングするか？(モデルを複雑にするほど較正は困難)

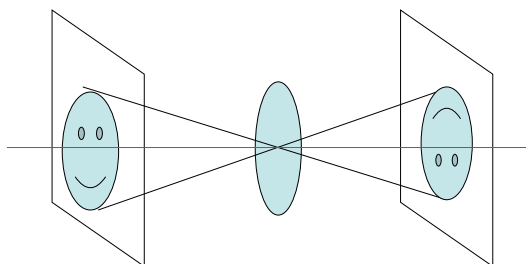


カメラの構造



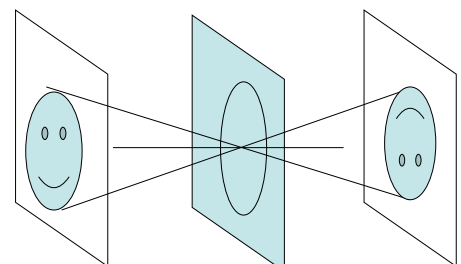
- レンズ - 集光するためのデバイス
 - 画像のぼけを無視した場合、無関係
 - 絞りを閉じた系で考える(主光線)

理想的な画像センサとは



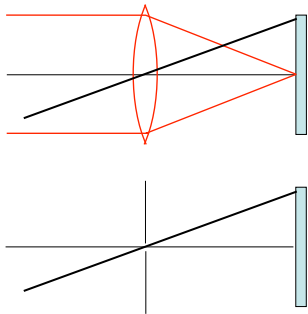
- 光軸(レンズの対称軸)に垂直な平面上の図形に対して、**相似の像**が得られること
 - ゆがみがあってはいけない
 - ぼけがあってはいけない

ピンホールカメラ



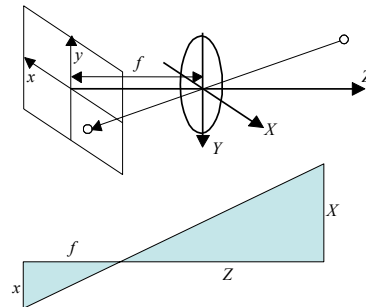
- 小さな「針穴」を通して像が出来る
 - 被写体と像の間には厳密な幾何学的関係が成立している(光の直進性より)
 - 実際には像が暗すぎてほとんど使われない

ピンホールカメラ



- 薄いレンズの中央を通る光は屈折されない
- レンズを絞り込んでいくとピンホールカメラに近づく

ピンホールカメラ



- ピンホール中央 (=前側レンズ主点) に原点を取る

$$\begin{cases} x = f \frac{X}{Z} \\ y = f \frac{Y}{Z} \end{cases}$$

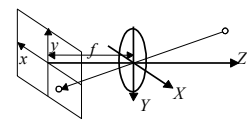
透視変換の同次座標表現

$$\begin{cases} x = f \frac{X}{Z} \\ y = f \frac{Y}{Z} \end{cases} \Leftrightarrow h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- 透視変換には除算が含まれる
 - 除算だけを最後まで「延期」して計算 (最後にhを消去)
 - 行列計算により透視変換を表現
 - 各ベクトルの末尾に要素“1”を追加する (同次座標表現)

透視変換に対する座標変換の導入

$$h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



- 上記の透視変換表現は様々な制限を有する
 - 世界座標の原点 = 投影中心 (レンズ主点)
 - 光軸 = Z軸に平行
 - 画像の中心 = 投影中心から下ろした垂線の足
 - アスペクト比 = 1.0
 - 座標変換を導入する必要あり

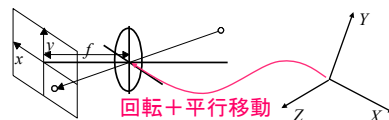
同次座標を用いた平行移動の表現

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \Leftrightarrow \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- 同次座標では積により平行移動が表現可能
 - $r_{11} \sim r_{22} \rightarrow$ 一次変換
 - $t_x \sim t_y \rightarrow$ 平行移動

世界座標系の導入

$$h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



- 透視変換行列と世界座標の間に、同次座標変換を挿入

- 外部パラメータ
 - 回転: 3自由度
 - 平行移動: 3自由度

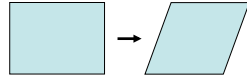
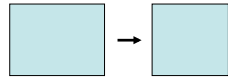
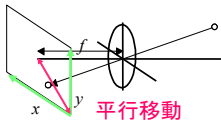
外部パラメータは、全部で6自由度!

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

外部パラメータ

挿入

内部パラメータの表現



- 内部パラメータ
 - 焦点距離 f (1自由度)
 - 画像中心 (2自由度)
 - アスペクト比 (1自由度)
 - スキュー歪み (1自由度)

内部パラメータは、
全部で5自由度!

内部パラメータの導入

- 内部パラメータは、透視変換行列の前に同次座標変換を掛ける

$$h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- a : アスペクト比
- s : スキュー比
- t_x, t_y : 画像中心

$$\begin{bmatrix} 1 & s & t_x \\ 0 & a & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

内部パラメータ

カメラパラメータ行列

$$h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & s & t_x \\ 0 & a & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

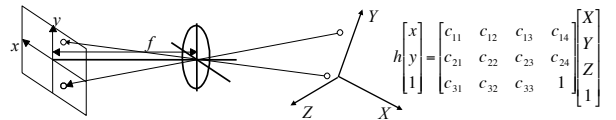
$$h \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- 行列の積をあらかじめ計算

- 3行4列の行列
カメラパラメータ

- パラメータ数 (自由度) は **11**
→ カメラパラメータ行列は定数倍しても意味が不変

基本的なキャリブレーション法



- 既知の $(X, Y, Z) \rightarrow (x, y)$ の組から較正
- カメラパラメータから h を消去

$$\begin{cases} hx = c_{11}X + c_{12}Y + c_{13}Z + c_{14} \\ hy = c_{21}X + c_{22}Y + c_{23}Z + c_{24} \\ h = c_{31}X + c_{32}Y + c_{33}Z + 1 \end{cases} \rightarrow \begin{cases} c_{31}Xx + c_{32}Yx + c_{33}Zx + x = c_{11}X + c_{12}Y + c_{13}Z + c_{14} \\ c_{31}Xy + c_{32}Yy + c_{33}Zy + y = c_{21}X + c_{22}Y + c_{23}Z + c_{24} \end{cases}$$

パラメータの計算

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -X_1x_1 & -Y_1x_1 & -Z_1x_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -X_1y_1 & -Y_1y_1 & -Z_1y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -X_nx_n & -Y_nx_n & -Z_nx_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -X_ny_n & -Y_ny_n & -Z_ny_n \end{bmatrix} \begin{bmatrix} c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \\ c_{21} \\ c_{22} \\ c_{23} \\ c_{24} \\ c_{31} \\ c_{32} \\ c_{33} \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ x_n \\ y_n \end{bmatrix}$$

- 未知数 11, 式 $2n$ (n : 特徴点数)

- 最小二乗法で解く。
上式を $Ax=y$ の形とすると

$$x = (A^T A)^{-1} A^T y$$

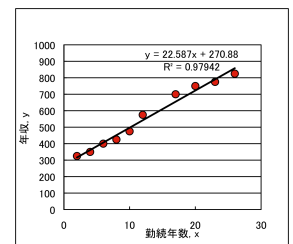
単回帰分析

1つの変数 x から、1つの変数 y を推定する。

例) 勤続年数と年収の関係を分析する。
直線で関係式を表現する。

$$y = ax + b \quad x: \text{説明変数} \\ y: \text{目的変数}$$

勤続年数, x	年収, y
2	325
4	350
6	400
8	425
10	475
12	575
17	700
20	750
23	775
26	825



最小二乗法

- モデル, データ
 - 回帰モデル $y = ax + b$
 - データ $(x_i, y_i), i = 1, \dots, N$
 - 規範
 - 残差平方和 $S = \sum (y_i - ax_i - b)^2$ を最小にする
 - 算法
 - S は a, b の二次式なので, $dS/da=0, dS/db=0$
- $$\frac{\partial S}{\partial a} = \sum_{i=1}^N 2(y_i - ax_i - b) \cdot (-x_i) = -\sum_{i=1}^N 2(y_i x_i - ax_i^2 - bx_i) = 0$$
- $$\frac{\partial S}{\partial b} = \sum_{i=1}^N 2(y_i - ax_i - b) \cdot (-1) = -\sum_{i=1}^N 2(y_i - ax_i - b) = 0$$

最小二乗法の行列解法

$$\sum_{i=1}^N y_i x_i = a \sum_{i=1}^N x_i^2 + b \sum_{i=1}^N x_i$$

$$\sum_{i=1}^N y_i = a \sum_{i=1}^N x_i + b \sum_{i=1}^N 1$$

$$\begin{bmatrix} \sum_{i=1}^N x_i^2 & \sum_{i=1}^N x_i \\ \sum_{i=1}^N x_i & \sum_{i=1}^N 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^N y_i x_i \\ \sum_{i=1}^N y_i \end{bmatrix}$$

共分散行列

$$ax_i + b = y_i$$

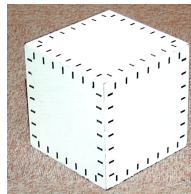
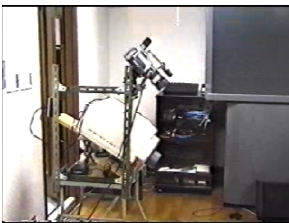
$$\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_N & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

$$X = (A^T A)^{-1} A^T B$$

Aの擬似逆行列

$$AX = B$$

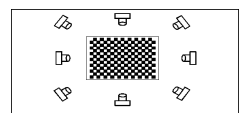
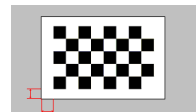
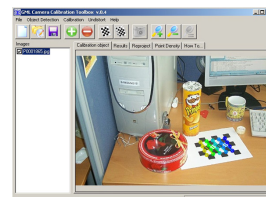
キャリブレーションの実際



キャリブレーションに用いる対象物体(一辺16cm)

キャリブレーションツール

- Z. Y. Zhang の手法
 - OpenCV ライブラリ (Intel)
 - GML C++ Camera Calibration toolbox



Calibration data: 21.10.2005 19:09:15
 Square size: 20.000 (mm)
 Board length: [2902.554 3000.251] [0.480 0.444]
 Principal point: [1134.889 824.027] [1.0 1.0] [0.480]
 Distortion: [4.219399 0.22649 0.001451 0.000205] [0.000506 0.000506 0.000000 0.000001]
 The camera matrix: [2902.554 0 1134.889 0 2000.251 824.027 0 0 1]
 Pixel size: [0.42 0.32]

カメラパラメータからの解析

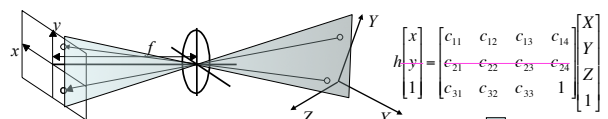
$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} f & s \cdot f & t_x \\ 0 & a \cdot f & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \Leftrightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

内部パラメータ
上3角行列

外部パラメータ
(左3x3)
正規直交行列

- カメラパラメータの左3x3を直交化
 - グラム・シュミットの正規直交化法を用いる
 - 内部パラメータとカメラの姿勢が求められる
 - 平行移動成分は内部パラメータ行列の逆行列を用いて求められる

カメラ2台の場合



- 一方のカメラのy座標は不要
 - y座標の行を削除して考える

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

カメラ1

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

カメラ2

三次元座標の算出

- 情報: スリット番号 x_p , 画素位置 x, y
- カメラパラメータ, プロジェクタパラメータを使用

$$F = \begin{bmatrix} x \cdot c_{34} - c_{14} \\ y \cdot c_{34} - c_{24} \\ x_p \cdot p_{24} - p_{14} \end{bmatrix} \quad Q = \begin{bmatrix} c_{11} - x \cdot c_{31} & c_{12} - x \cdot c_{32} & c_{13} - x \cdot c_{33} \\ c_{21} - y \cdot c_{31} & c_{22} - y \cdot c_{32} & c_{23} - y \cdot c_{33} \\ p_{11} - x_p \cdot p_{21} & p_{12} - x_p \cdot p_{22} & p_{13} - x_p \cdot p_{23} \end{bmatrix}$$

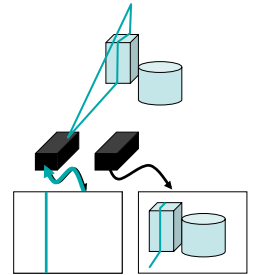
- より

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Q^{-1} \cdot F$$

で座標が求められる

能動型ステレオ法

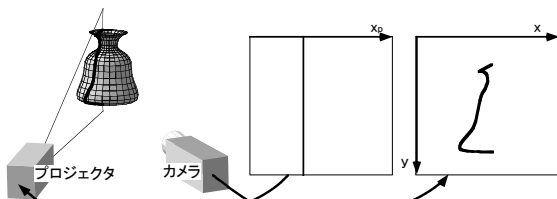
- 一方のカメラをプロジェクタに置き換える(光を投影)
- もう一方のカメラでその光パターンの像を計測
- 対応付け問題が簡単に
- 模様のない物体でも計測可能
- パターン光を投影しながら画像を取り込む



→ 画像の取り込み回数が多く, 時間がかかる

→ ハードウェアによる高速化・投影光の工夫等

応用: レンジファインダ



- スリット光投影法
- プロジェクタはカメラ同様にモデル化可能

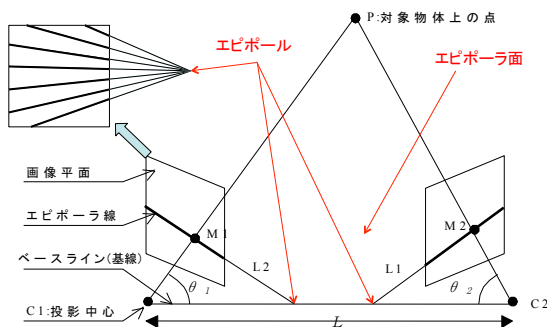
プロジェクタのモデル化

- スリット光プロジェクタは1次元表示デバイス
- y は任意の値をとるため, y 成分を省く

$$h \begin{bmatrix} x_p \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- カメラと同様にキャリブレーション可能
- どの位置のスリット光が, どの座標に到達するか

エピポーラ幾何の用語



F行列

$$m_1 = \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

$$m_2 = \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x_1 & y_1 & 1 \end{bmatrix} \cdot F \cdot \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = 0$$

(x_2, y_2) を決めると $ax_1 + by_1 + c = 0$
 (x_1, y_1) を決めると $ax_2 + by_2 + c = 0$

$m_1^T F m_2 = 0$ F はカメラパラメータから算出可能