


# 画像メディア工学特論(5)

対象物体の追跡


# 2010年PRMUアルコン



第14回 パターン認識・メディア理解研究会 (PRMU)  
アルゴリズムコンテスト

## ターゲットをロックオンせよ!

～移動物体の追跡～



[http://www.murase.m.is.nagoya-u.ac.jp/  
alcon2010/](http://www.murase.m.is.nagoya-u.ac.jp/alcon2010/)

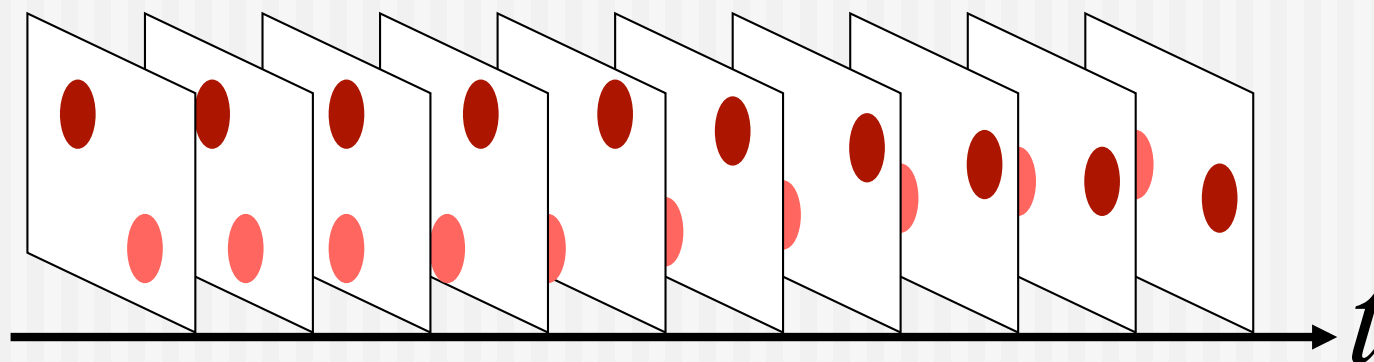
- コンテストHP, サンプルをよく見てみる
- 気づいたことを, 次回報告
  
- (好みの, 手持ちの) プログラミング環境は?

# 今回の内容

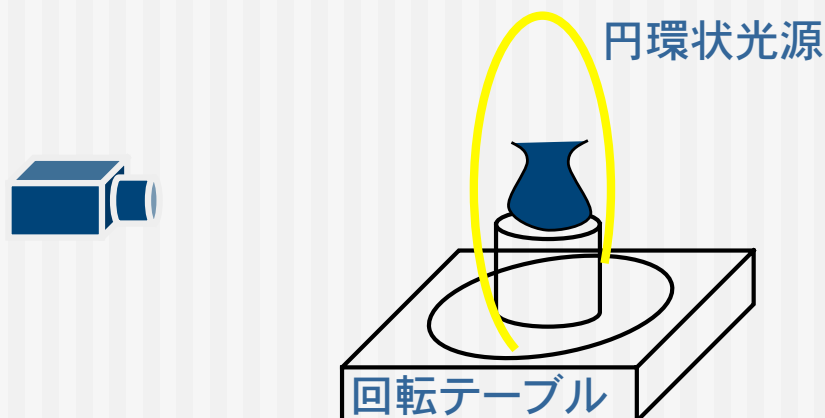
- 「対象追跡」に関する講義
- サーベイ論文をもとに進行
  - 加藤丈和, 深尾隆則, 羽下哲司, 「対象追跡 - フレーム間の類似度に着目した手法から動きのモデルに着目した手法まで」, 情報処理学会研究報告 2005-CVIM-150 (2005年9月)

# 動画像の定義とは？

- 時系列に得られた一連の画像群



- 撮影パラメータを変更しながら得た画像群は？



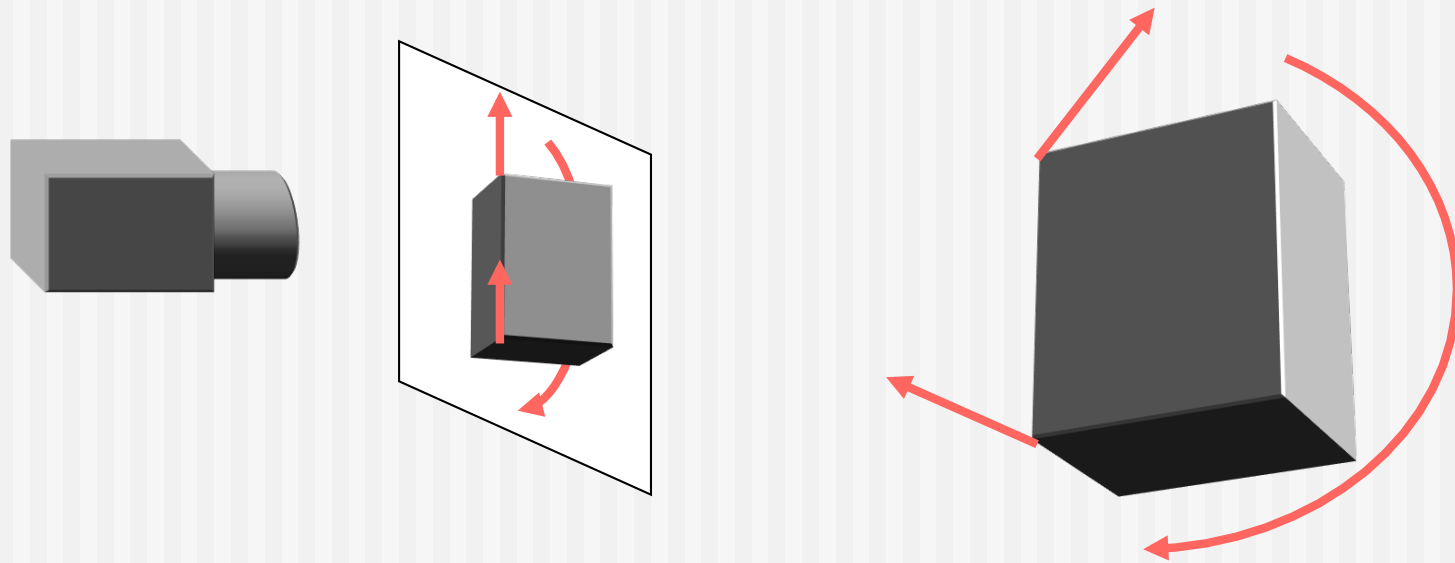


# 動画像の種類

- 対象が運動
  - カメラが運動
    - カメラの撮影方位変化
    - カメラの位置変化
  - 照明条件の変化
  - 撮影パラメータの変化
    - ズーム, フォーカス等
- 
- 同一視可能
- 性質近い

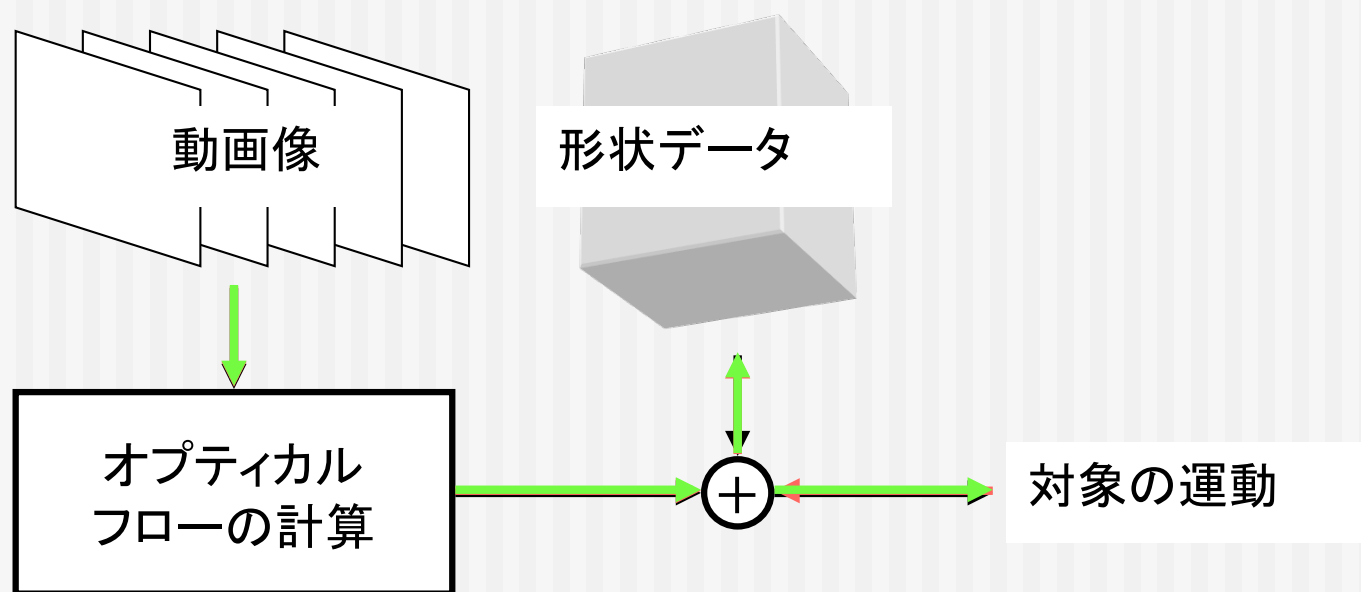
# オプティカルフロー

---



- 3次元的な動きを画像平面に投影したもの
  - 実際の物体の動きは求められない

# 動画像処理の流れ



- 形状計測
- 運動計測
- 形状と運動の同時推定

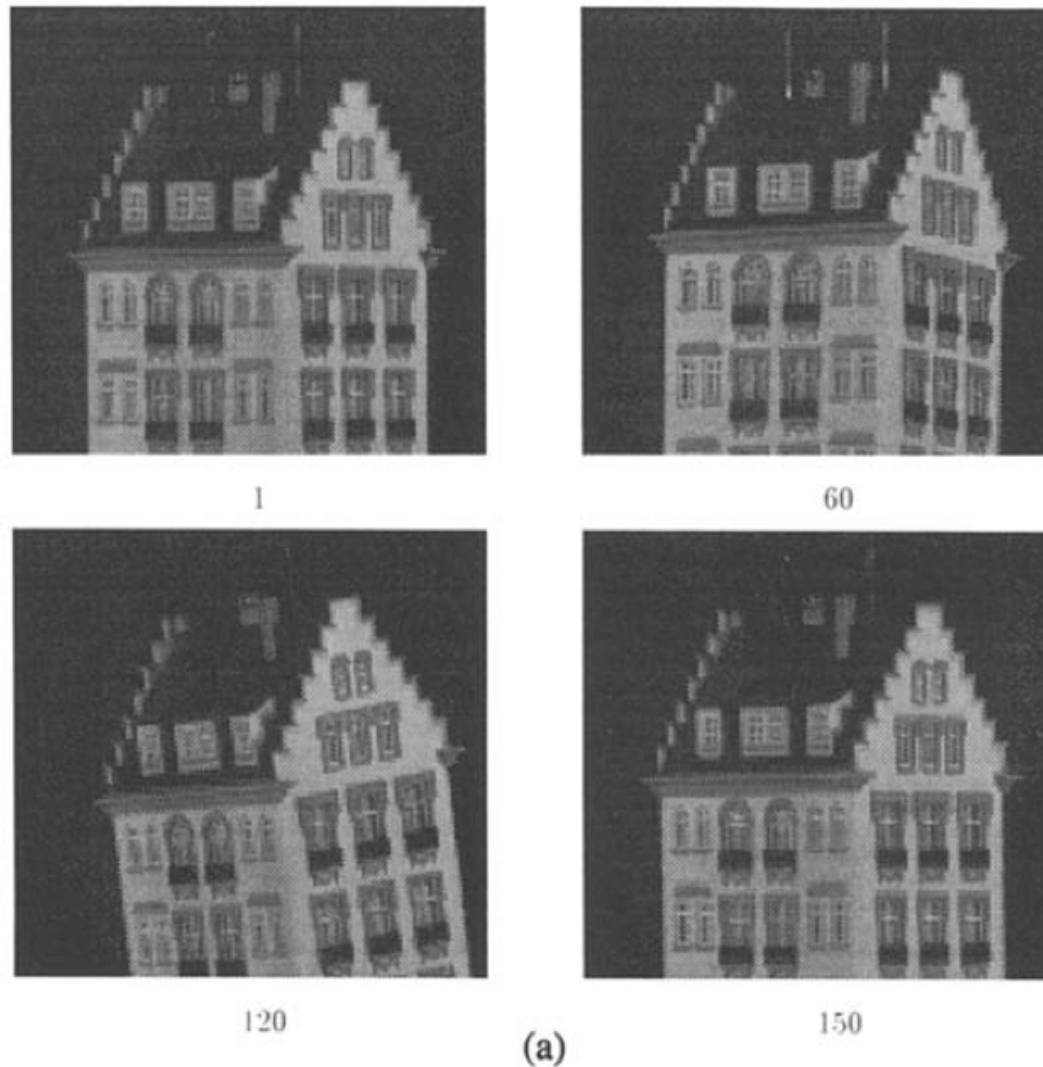
# 対象追跡(1) 特徴点追跡



- 2枚の画像から、対応する点の対応を探す

# 特徴点追跡の応用

入力画像

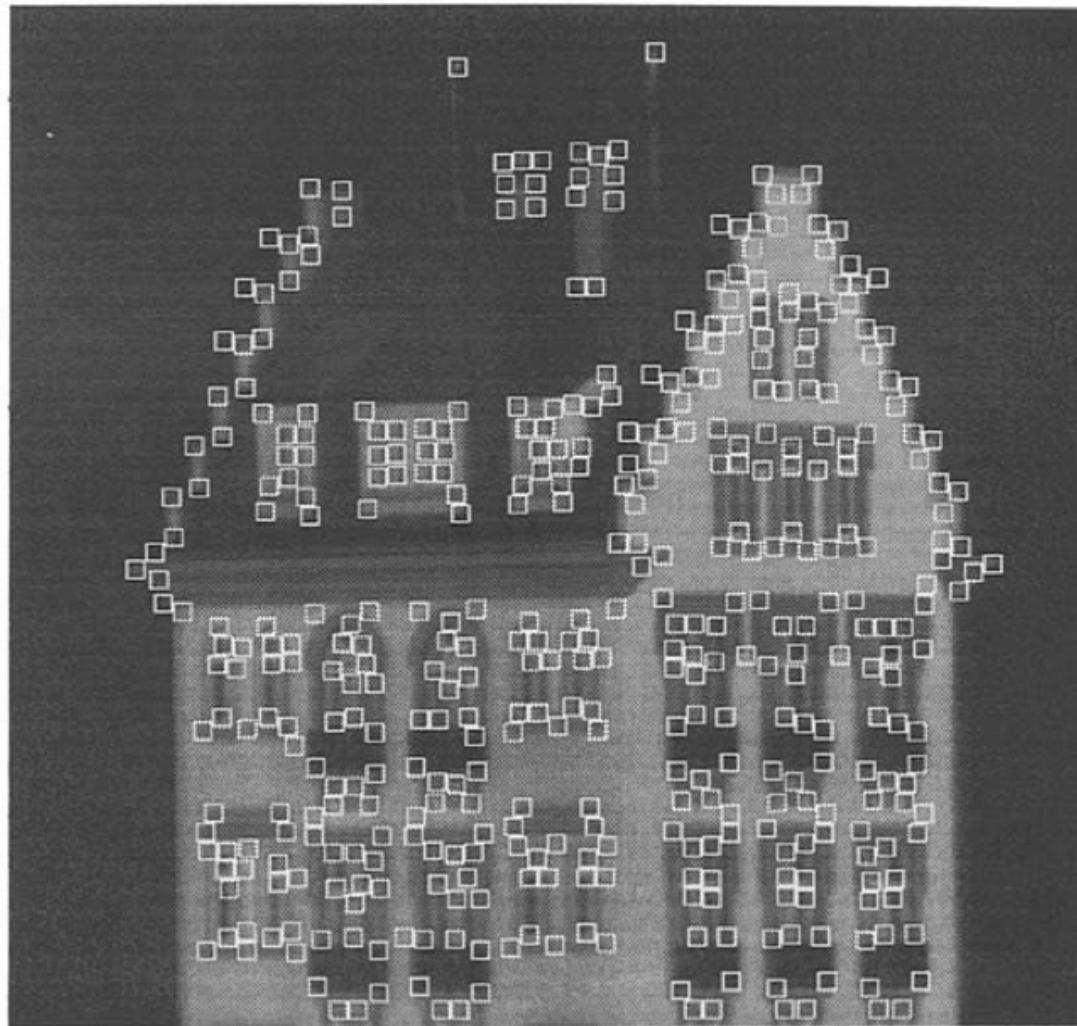


*Fig. 2a.* The “Hotel” stream: four of the 150 frames.

Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137–154.

# 特徴点追跡の応用

追跡され  
た特徴点



(b)

Fig. 2b. The "Hotel" stream: the 430 features selected by the automatic detection method.

Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137-154.



# 特徴点追跡の応用

推定された形状

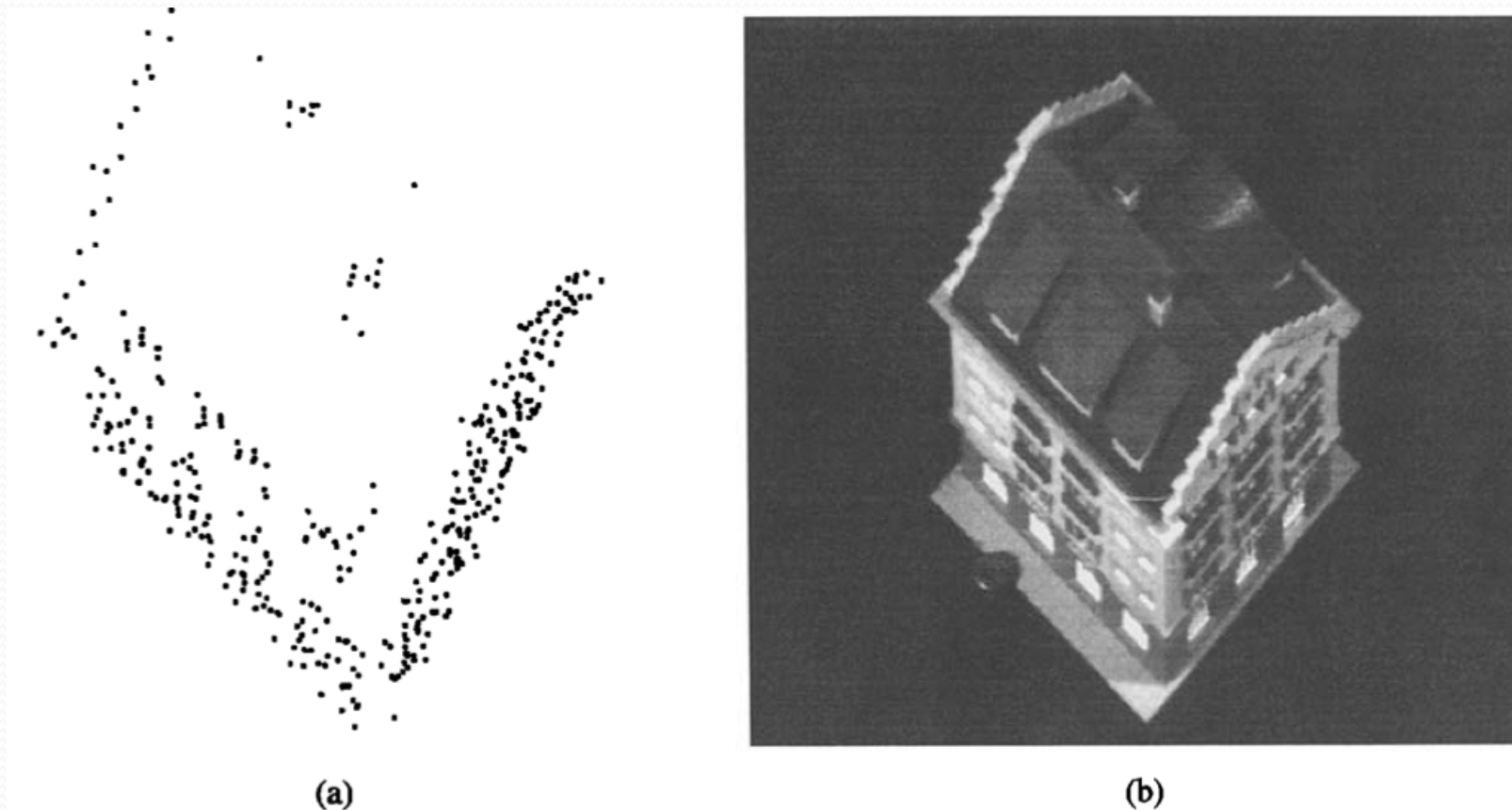


Fig. 4. Qualitative shape results for the "Hotel" stream: top view of the (a) computed and (b) actual shape.

Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137-154.

# 特徴点追跡の応用

## 入力画像と特徴点



40



80

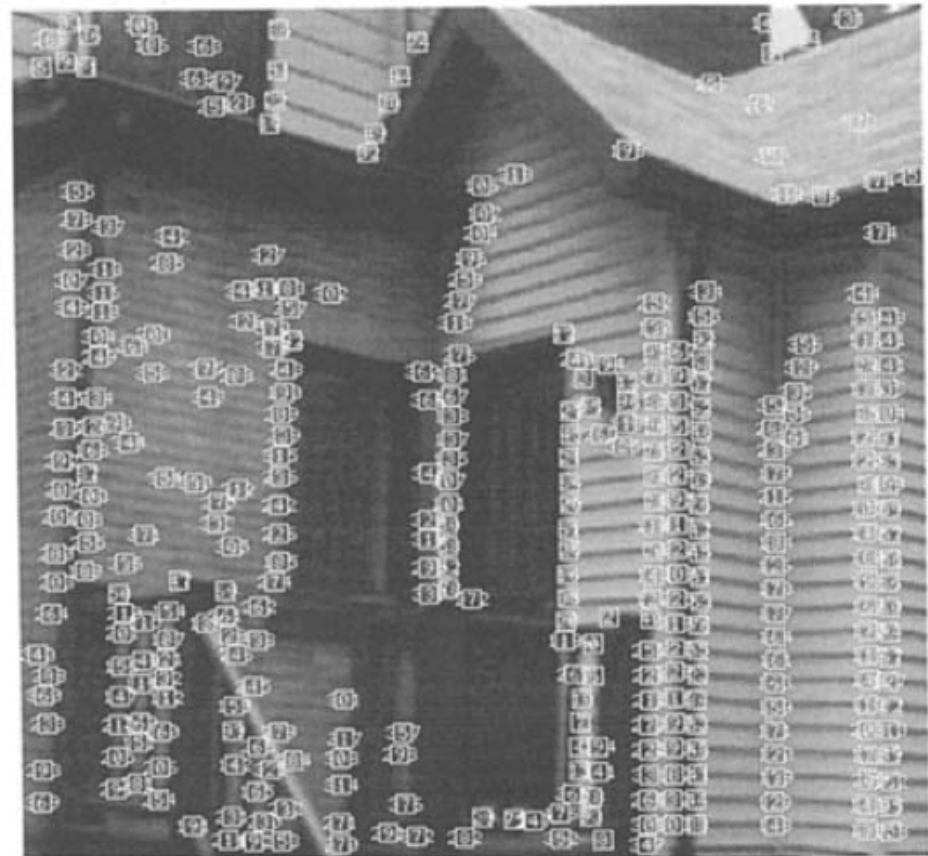


110



180

(a)



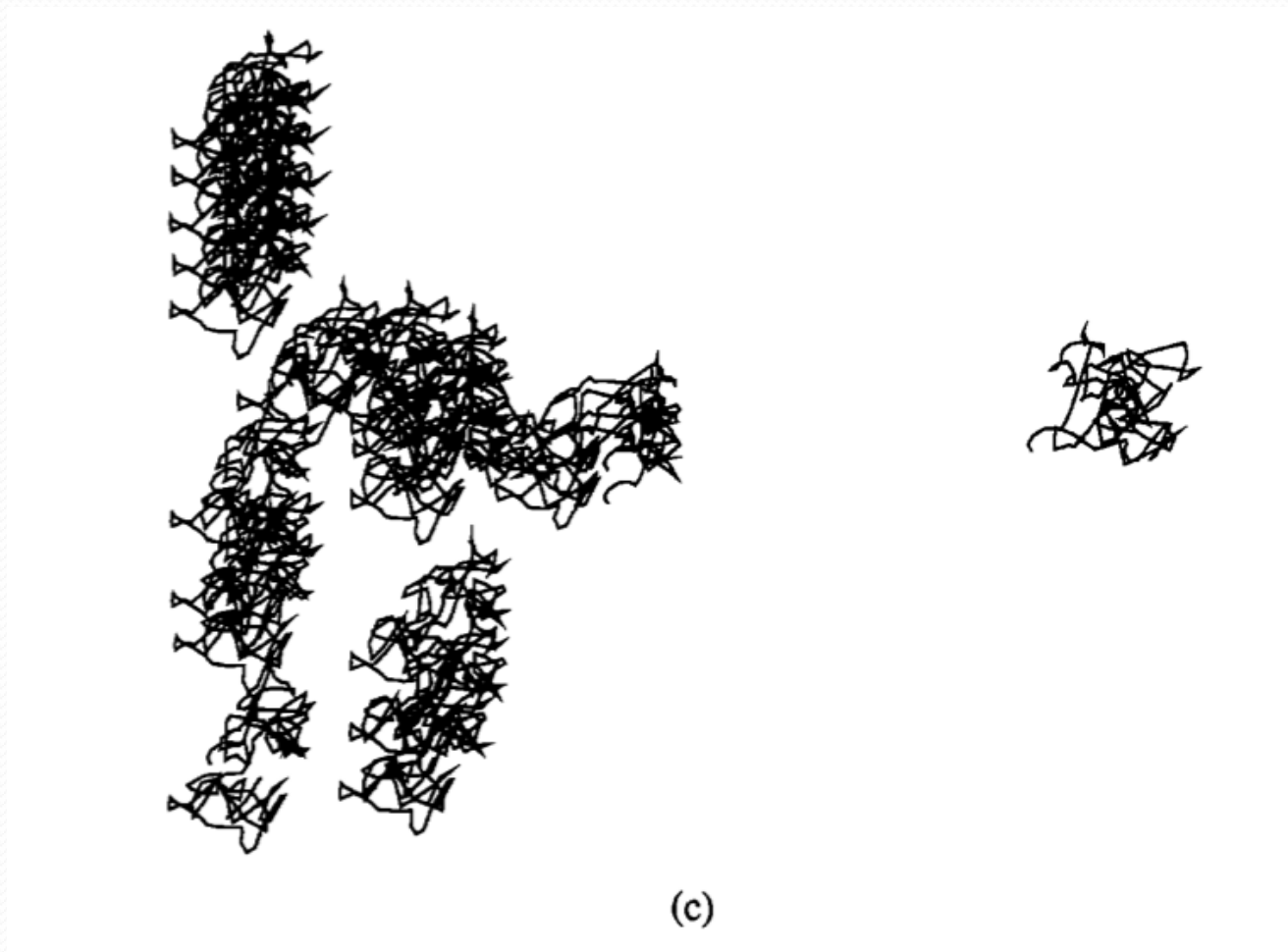
(b)

Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137-154.



# 特徴点追跡の応用

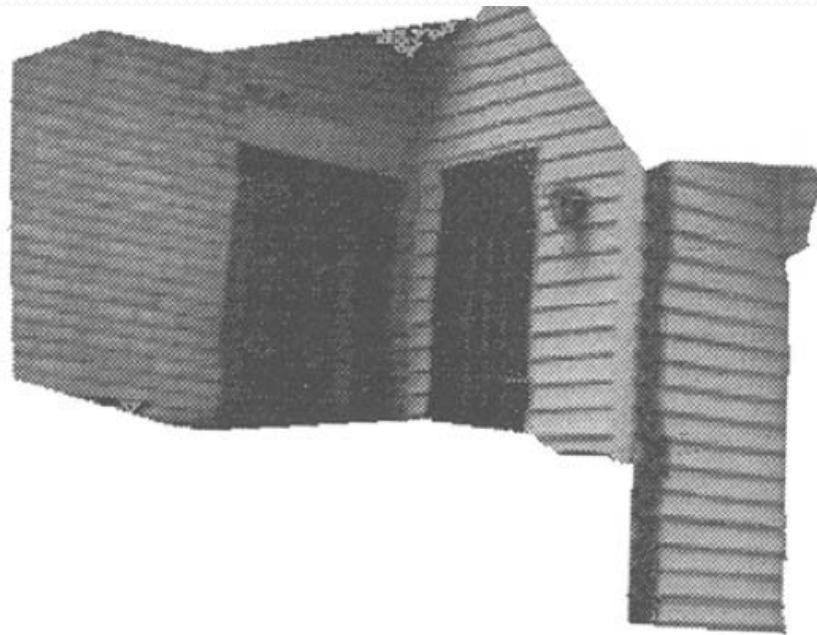
特徴点の軌跡



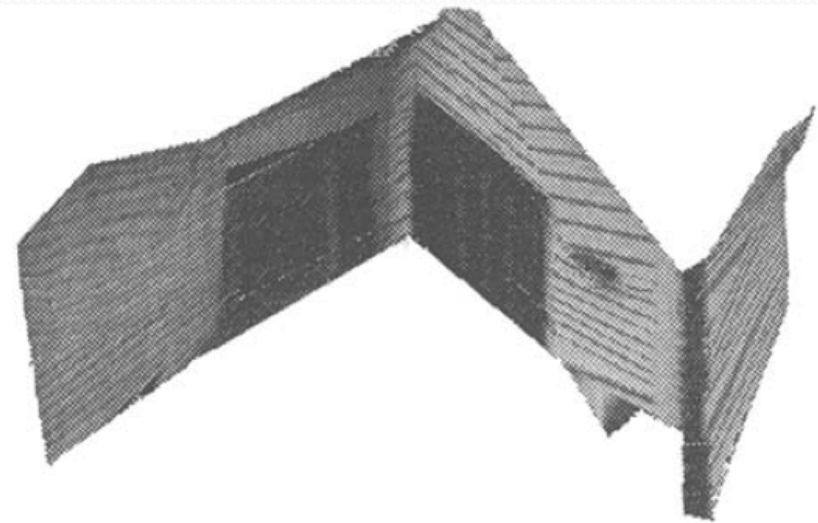
Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137-154.

# 特徴点追跡の応用

形状をもとにレンダリングした結果



(a)



(b)

Carlo Tomasi and Takeo Kanade. (November 1992.). "Shape and motion from image streams under orthography: a factorization method.". *International Journal of Computer Vision*, 9 (2): 137-154.

# 特徴点追跡の応用

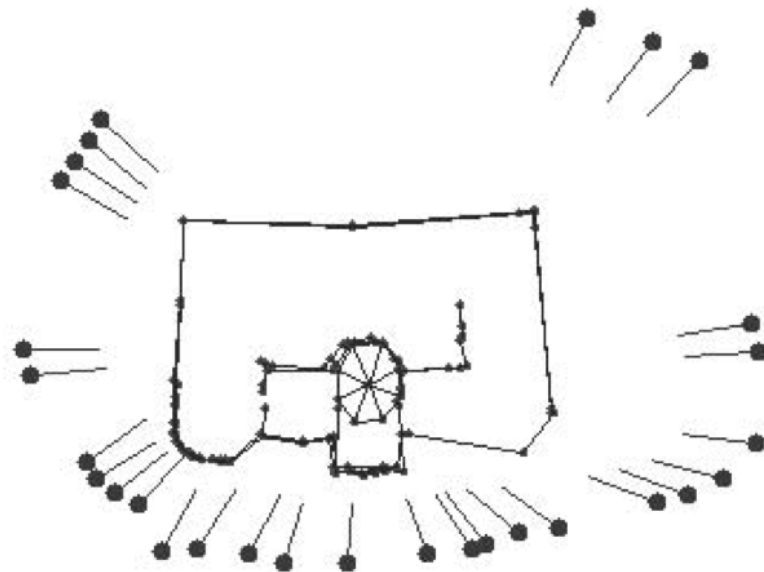


図 10 原画像の例と復元結果  
原爆ドーム ( $P = 122$ ,  $F = 29$ ,  $r = 17\%$ )

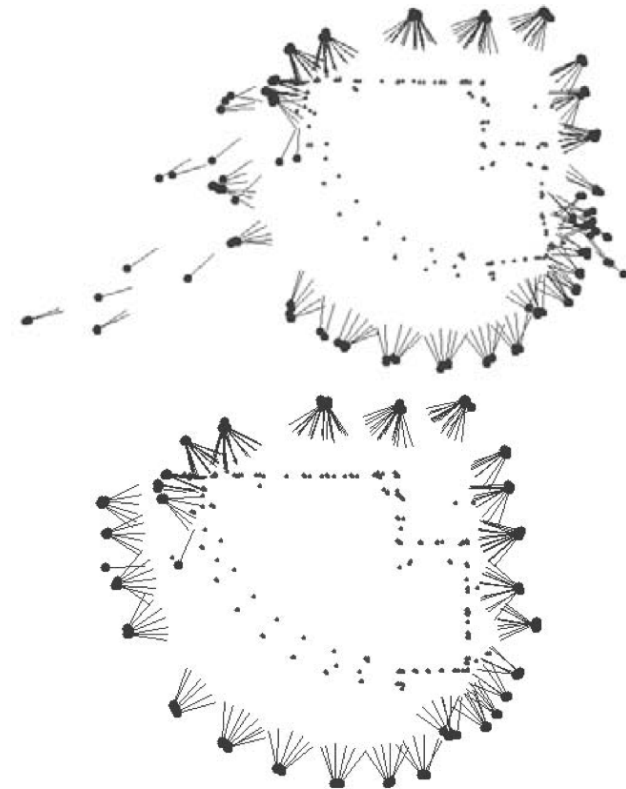


図 11 原画像の一部と中間結果, 復元結果  
講堂 ( $P = 189$ ,  $F = 280$ ,  $r = 6\%$ )

椋木雅之, 右田剛史, 青山正人, 浅田尚紀, 非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法, 情報処理学会論文誌: コンピュータビジョンとイメージメディア, Vol. 45, No. SIG 13(CVIM 10), pp. 64-73, 2004.

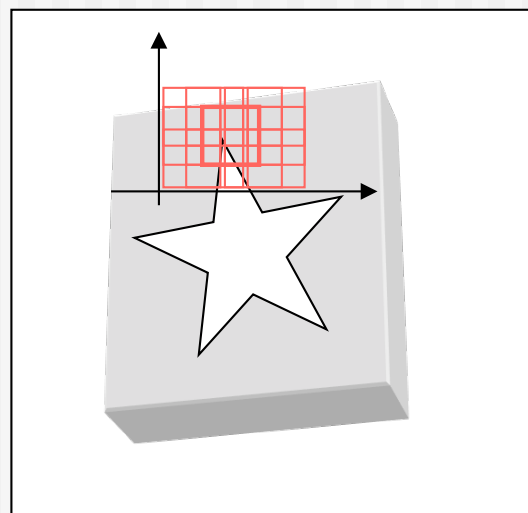
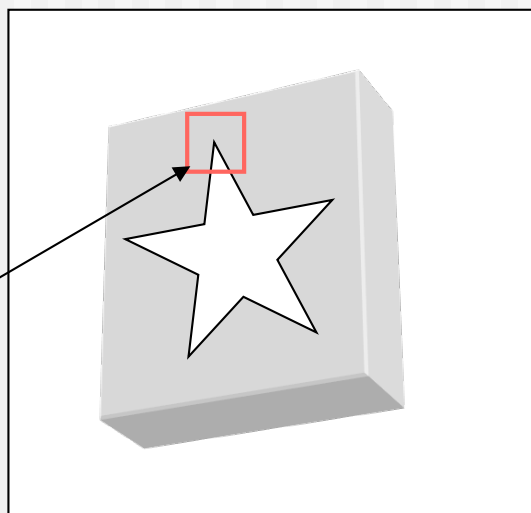
# オプティカルフローの推定

---

- 動画像において、対象の画像上の動きを各画素ごとに求める
  - ブロックマッチング
  - 濃度勾配法
  - 時空間画像解析
    - EPI(エピポーラ平面画像)
    - 時空間FFT

# ブロックマッチング法

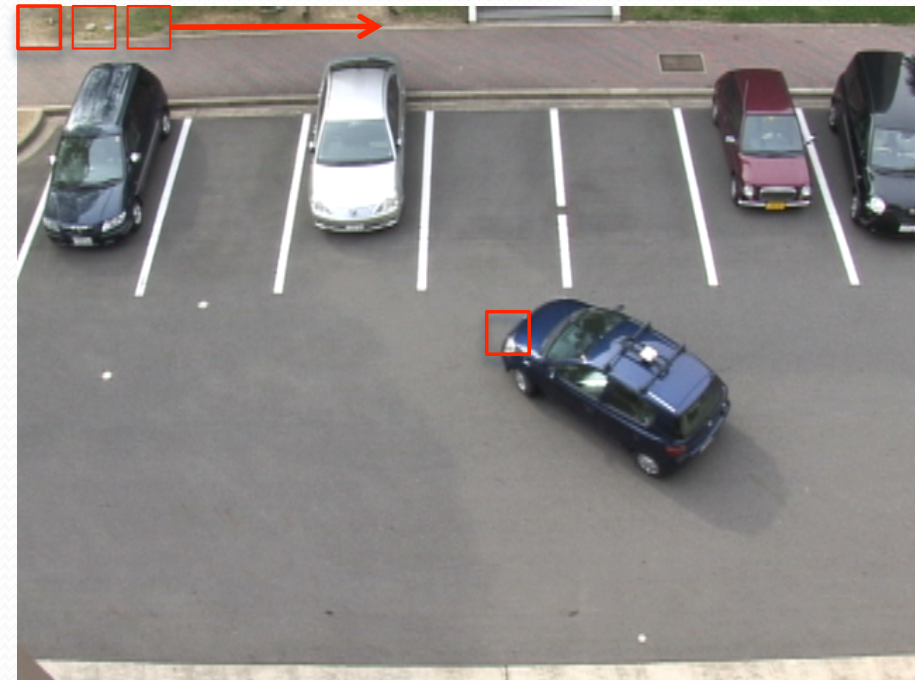
テンプレート



- 前後の画像でテンプレートマッチングを行う
  - 計算時間がかかる → 専用ハードウェアの出現
  - ブロックの変形・ブロック内の複数物体に弱い



# 特徴点追跡の方法



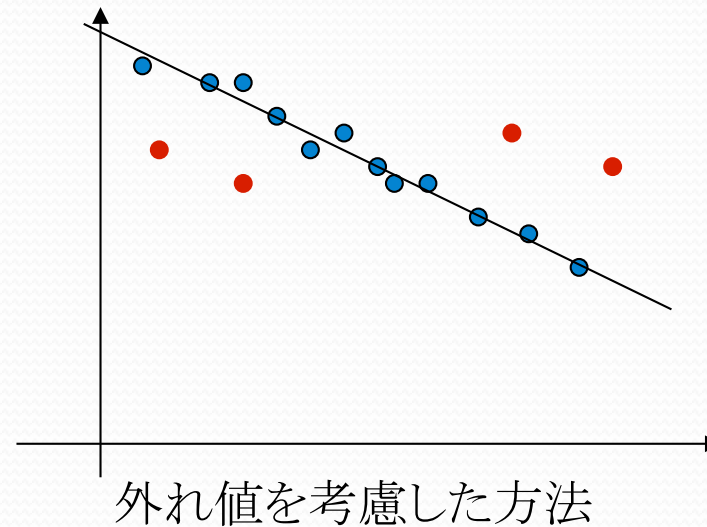
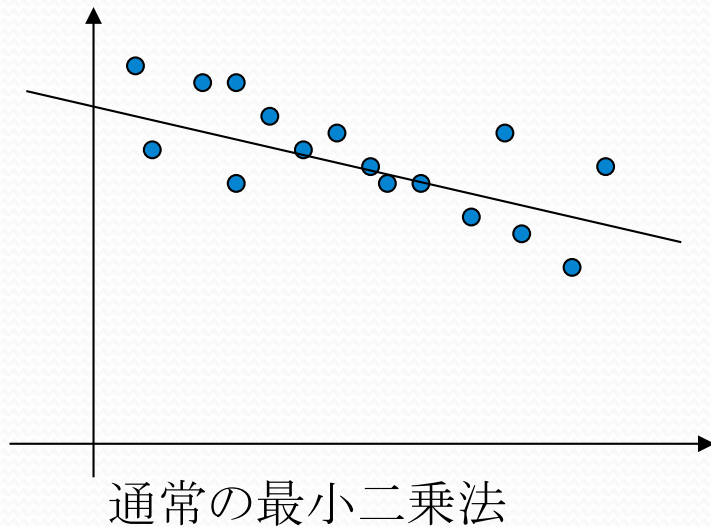
- ブロックマッチング, テンプレートマッチング
  - 画像の一部を切り出して, 比較する
    - 比較 (差の計算方法) は, どうするか?
    - ブロックサイズ (ウィンドウサイズ) をどうするか?
    - 探索戦略をどうするか?

# ブロックマッチングの計算方法

- SSD (Sum of Squared Differences)
  - 画素値の差の二乗和を用いる (最小二乗)  
2乗誤差和を最小化する変位量  $d$ を画像  $J$  にて発見する方法である。

$$\epsilon = \int_w [J(\mathbf{x} + \mathbf{d}) - I(\mathbf{x})]^2 d\mathbf{x} \quad (1)$$

- 外れ値に弱い
- 外れ値とは？



# ブロックマッチングの計算方法

- SAD (Sum of Absolute Differences)
  - 画素値の差の絶対値の和を用いる

SSD は外れ値 (Outlier) が存在するときに、大きな影響を受けるため、単純に以下のような絶対値の誤差和を用いる方法も考えられている。

$$\epsilon = \int_w |J(\mathbf{x} + \mathbf{d}) - I(\mathbf{x})| dx \quad (2)$$

- SSD, SAD は全体的な輝度変化に弱い





# ブロックマッチングの計算方法

- NCC (Normalized Cross Correlation)
  - 相互相関 (内積と同じ)

さらに, SSD から導かれる指標として以下の相互相関を用い, 最大化する変位量  $\mathbf{d}$  を求める方法も高速な実装が可能なものとして用いられている.

$$\gamma = \int_{\mathcal{W}} [I(\mathbf{x}) - \bar{I}][J(\mathbf{x} + \mathbf{d}) - \bar{J}] d\mathbf{x} \quad (3)$$

ここで,  $\bar{I}$  は領域  $\mathcal{W}$  における  $I$  の平均,  $\bar{J}$  は  $J$  の平均を表す.

- 正規相互相関 (正規化相互相関)
  - 長さ 1 にしたベクトルの内積, つまり, なす角度の  $\cos$

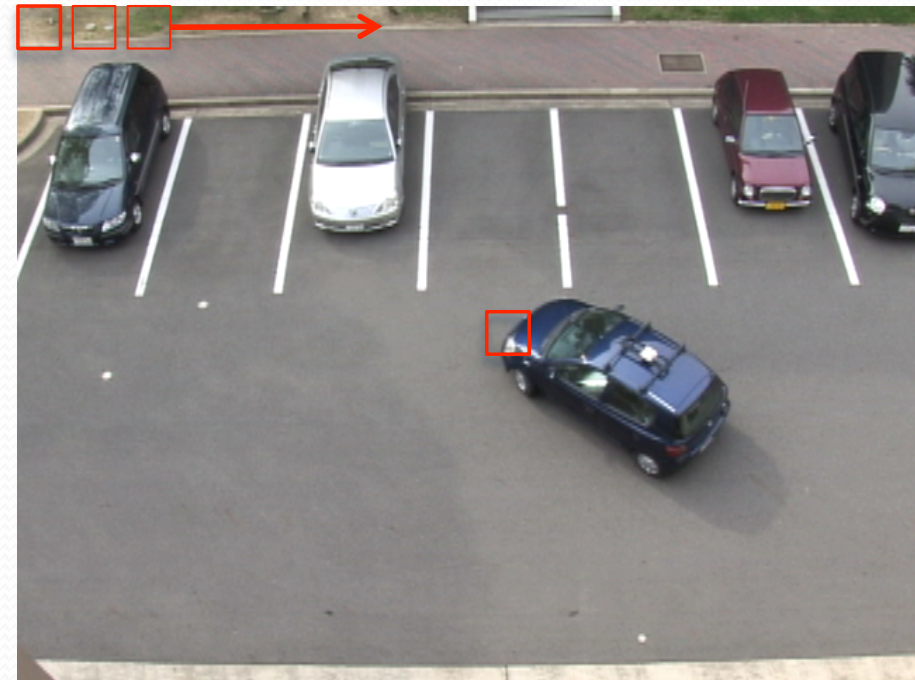
また, 特に照明変化のある時などに, 次の正規相互相関 (Normalized Cross Correlation) を指標として用いる方法 [2] も一般的である.

$$\gamma = \frac{\int_{\mathcal{W}} [I(\mathbf{x}) - \bar{I}][J(\mathbf{x}) - \bar{J}] d\mathbf{x}}{\sqrt{\int_{\mathcal{W}} [I(\mathbf{x}) - \bar{I}]^2 d\mathbf{x} \int_{\mathcal{W}_1} [J(\mathbf{x}) - \bar{J}]^2 d\mathbf{x}}} \quad (4)$$

# ブロックマッチングの計算量

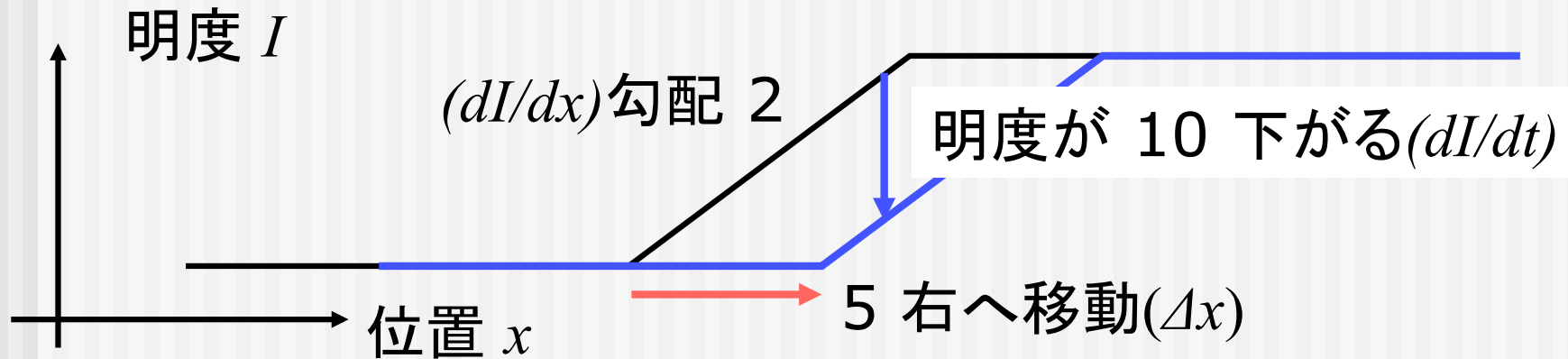
- 力任せの方法
  - 探索したいブロックの個数
    - 画像の画素数, または画素数/ウィンドウサイズ
  - 探索回数
    - 画像の画素数 (1画素の精度で移動を求めるとき)
  - 1回のマッチングの計算量
    - ウィンドウサイズ (ウィンドウ内の画素数)
    - SSD, SAD, NCC で同じオーダー
- 非常に計算量が多い
  - 現在の計算機でも数分～数十分を要する

# 探索戦略



- フレーム間の移動量が十分小さいとき
  - 探したいブロックの近くに対応点が存在する可能性が高い
  - 周りのブロックと似た向きに動いている可能性が高い
  - 前回の移動と似た運動がつづいている可能性が高い

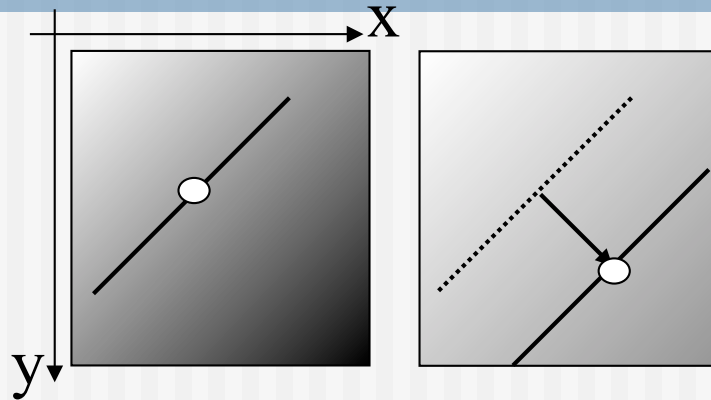
# 濃度勾配法(1)



- 移動量  $\times$  勾配 = 明度変化量  
 $\Delta t$  時間あとに  $\Delta x$  だけ移動したとすると

$$\Delta x \cdot \frac{\partial I}{\partial x} = -\Delta t \cdot \frac{\partial I}{\partial t}$$

# 濃度勾配法(2)



$$I(x, y, t) =$$

$$I(x + \Delta x, y + \Delta y, t + \Delta t)$$

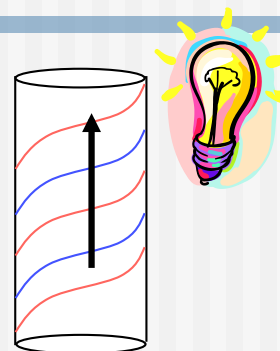
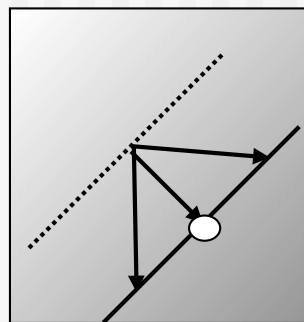
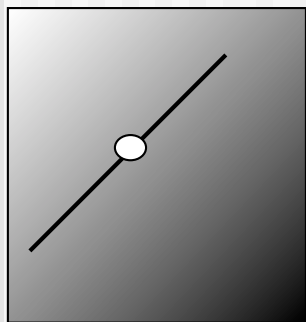
- テイラー展開により

$$I(x, y, t) = I(x, y, t) + \Delta x \frac{\partial I}{\partial x} + \Delta y \frac{\partial I}{\partial y} + \Delta t \frac{\partial I}{\partial t} + \varepsilon$$

- 高次誤差を無視すると

$$\Delta x \frac{\partial I}{\partial x} + \Delta y \frac{\partial I}{\partial y} = -\Delta t \frac{\partial I}{\partial t}$$

# アパーチャ問題

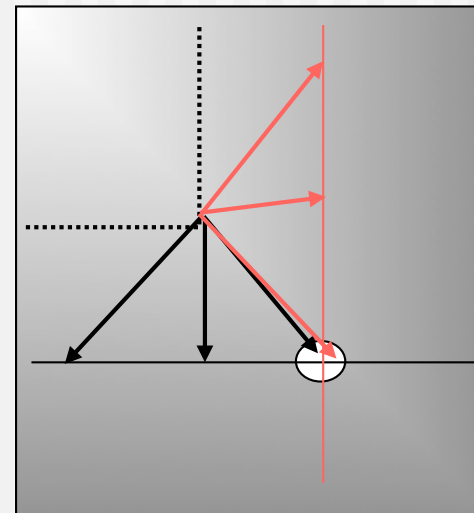
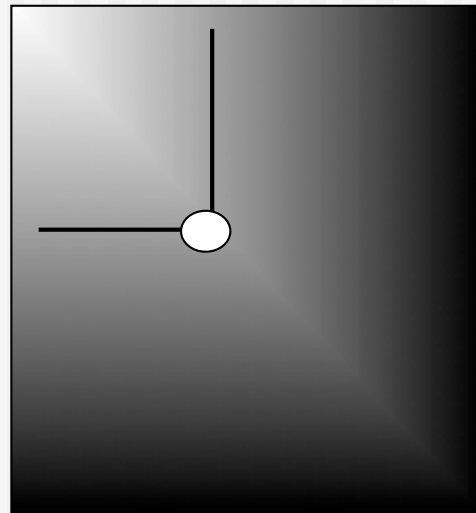


- 式1つ, 未知数2つ → 不確定性

$$\Delta x \frac{\partial I}{\partial x} + \Delta y \frac{\partial I}{\partial y} = -\Delta t \frac{\partial I}{\partial t}$$

- $\Delta x, \Delta y$  はある直線上に拘束される

# アパーチャ問題の解決



- 動きの「なめらかさの拘束」を用いる
  - 近傍の動きが同じであるという仮定
- 近傍で求められた拘束線の交点を用いる

# Horn-Schunckの方法

- 滑らかさの拘束を用いて、画像全体でオプティカルフローを求める方法
  - B.K.P.Horn and B.G. Schunck Determining Optical Flow, AI(17), No. 1-3, 1981, pp. 185-203

$$\frac{dE}{dt} = 0.$$

Using the chain rule for differentiation we see that,

$$\frac{\partial E}{\partial x} \frac{dx}{dt} + \frac{\partial E}{\partial y} \frac{dy}{dt} + \frac{\partial E}{\partial t} = 0.$$

(See Appendix A for a more detailed derivation.) If we let

$$u = \frac{dx}{dt} \quad \text{and} \quad v = \frac{dy}{dt},$$

then it is easy to see that we have a single linear equation in the two unknowns  $u$  and  $v$ ,

$$E_x u + E_y v + E_t = 0,$$



# Horn-Schunckの方法

## 5. The Smoothness Constraint

One way to express the additional constraint is to minimize the square of the magnitude of the gradient of the optical flow velocity:

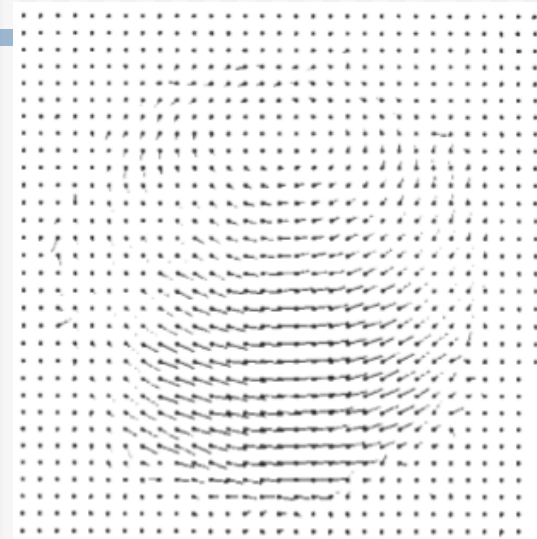
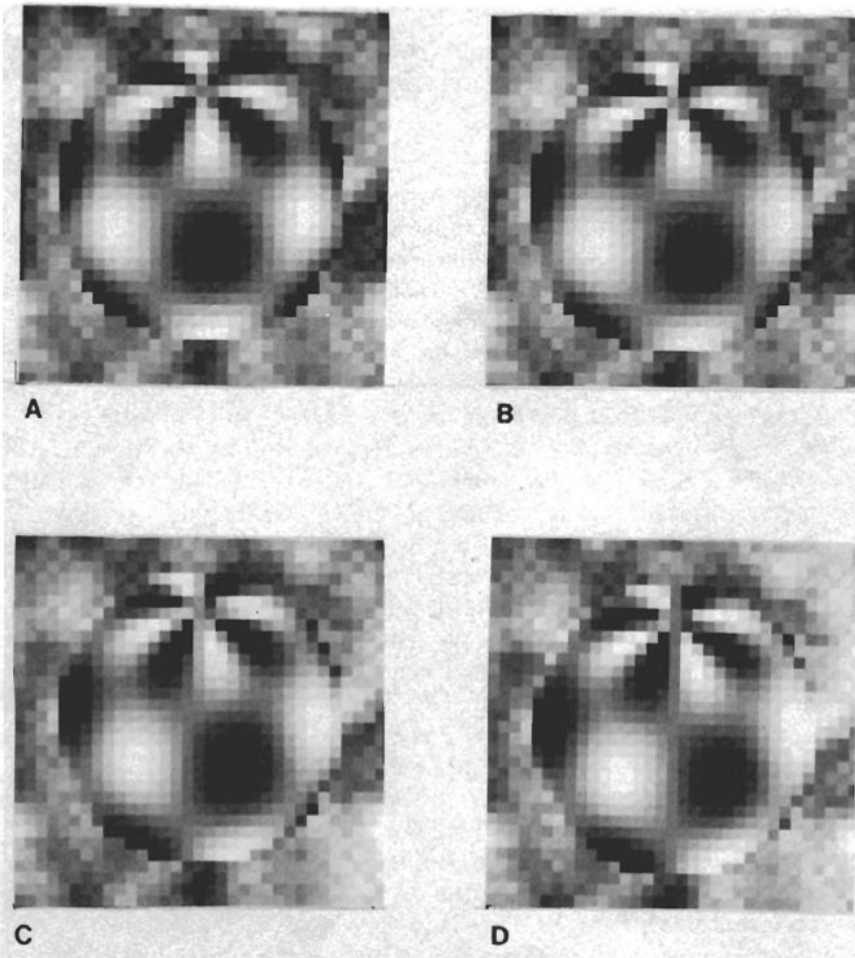
$$\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 \quad \text{and} \quad \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2.$$

Another measure of the smoothness of the optical flow field is the sum of the squares of the Laplacians of the  $x$ - and  $y$ -components of the flow. The Laplacians of  $u$  and  $v$  are defined as

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad \text{and} \quad \nabla^2 v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}.$$

In simple situations, both Laplacians are zero. If the viewer translates parallel to a flat object, rotates about a line perpendicular to the surface or travels orthogonally to the surface, then the second partial derivatives of both  $u$  and  $v$  vanish (assuming perspective projection in the image formation).

# Horn-Schunckの方法

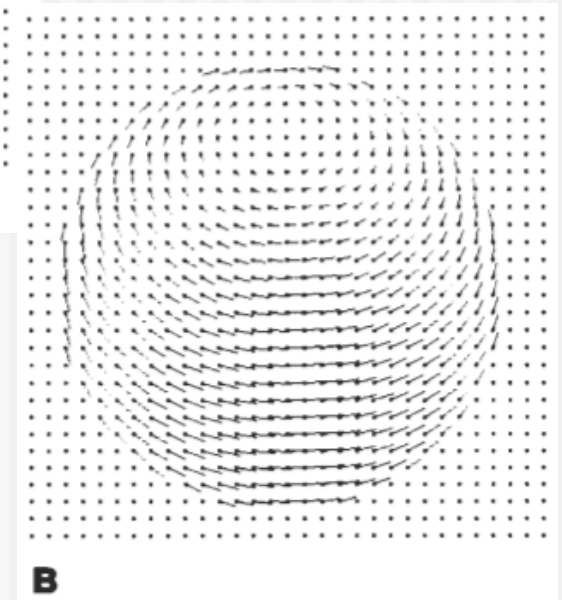


推定値



入力

正解



B

B.K.P.Horn and B.G. Schunck Determining Optical Flow, AI (17), No. 1-3, 1981, pp. 185-203

# Lucas-Kanade の方法

---

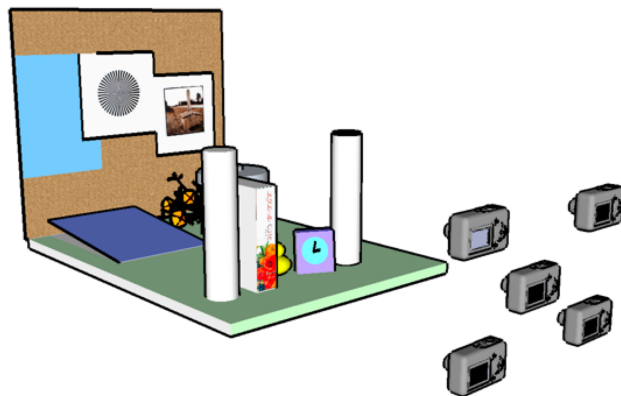
- 勾配法を, ウィンドウ内の画素で連立させて解く方法
  - 配布資料「対象追跡 - 」では, ウィンドウの平行移動, 回転, 大きさの変化に対応した手法について述べられている
  - 単純な平行移動について, 以下がわかりやすい
    - [http://en.wikipedia.org/wiki/Lucas-Kanade\\_Optical\\_Flow\\_Method](http://en.wikipedia.org/wiki/Lucas-Kanade_Optical_Flow_Method)
- 特徴点選択を組み合わせたKLT(Kanade-Lucas-Tomasi)法がよく使われている
  - OpenCV ライブラリにもある



# 補間を用いたぼけ画像作成実験



入力画像



目標出力画像

大きな口径のカメラを用いて撮影した  
大きなぼけを含む画像

# 撮影画像の重ね合わせによる不連続なぼけ

焦点面が合うように重ね合わせ



変形画像の重ね合わせ(前景に焦点)

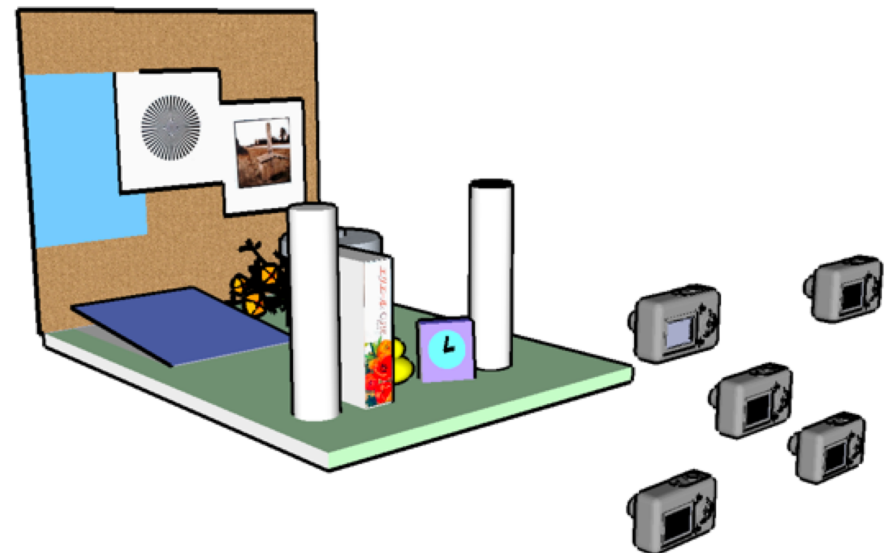
○ 合焦面が鮮明

## ■ 手持ちによる撮影の問題 2

ぼけ部分に輪郭線が出現



撮影点の間隔が広すぎる



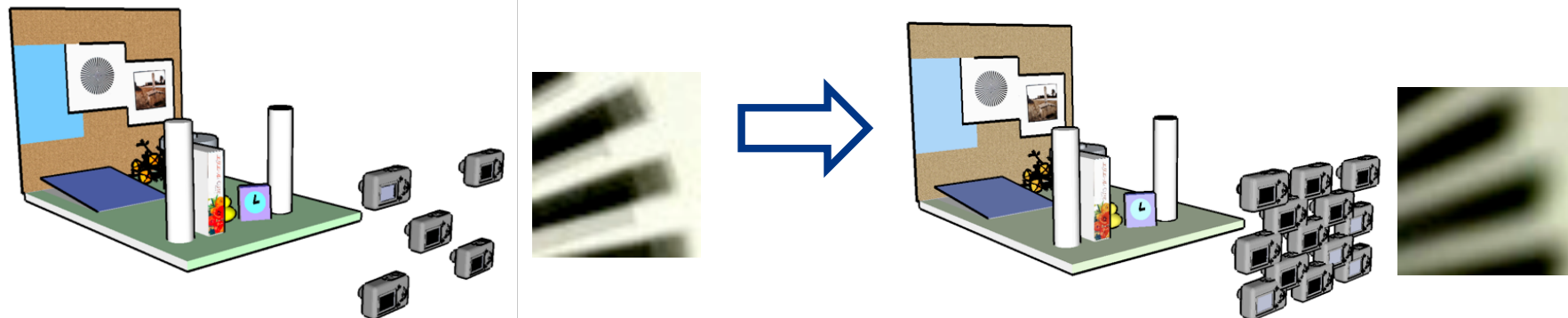
撮影点の位置関係



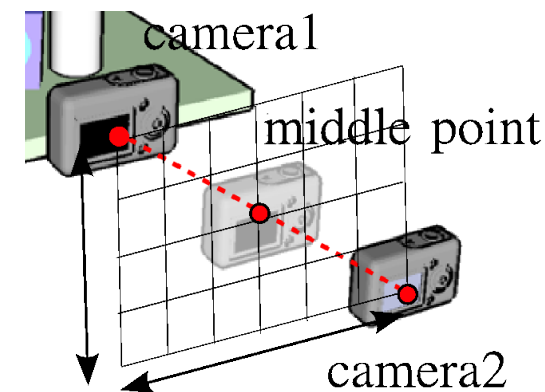
# 補間画像によるぼけ部分の円滑化

## ■ 撮影画像間の中間画像を作成

- 中間画像で補間することで滑らかなぼけを実現  
= 疎な撮影点を密に補間



- 撮影画像間の視差から撮影位置関係を推定
  - ・ 補間画像の相対的な撮影点を指定
- 撮影画像から補間画像を作成
  - ・ モーフィング技術を採用



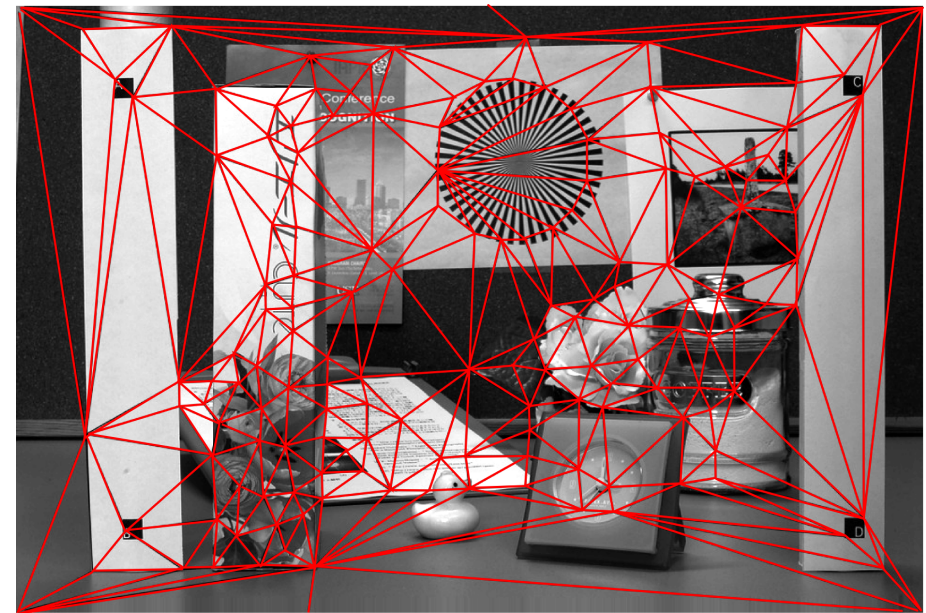
# 中間画像の作成過程

- 得られる視差は追跡可能な特徴点についてのみ
  - 特徴点以外の画素の視差を補間によって求める

特徴点同士を結んで領域分割  
分割した小領域ごとに線形補間



画像間の視差の計算結果



中間画像の作成過程（領域分割）



# 補間を用いたぼけ画像作成結果



前景に焦点



後景に焦点

撮影画像5枚  
補間画像100枚

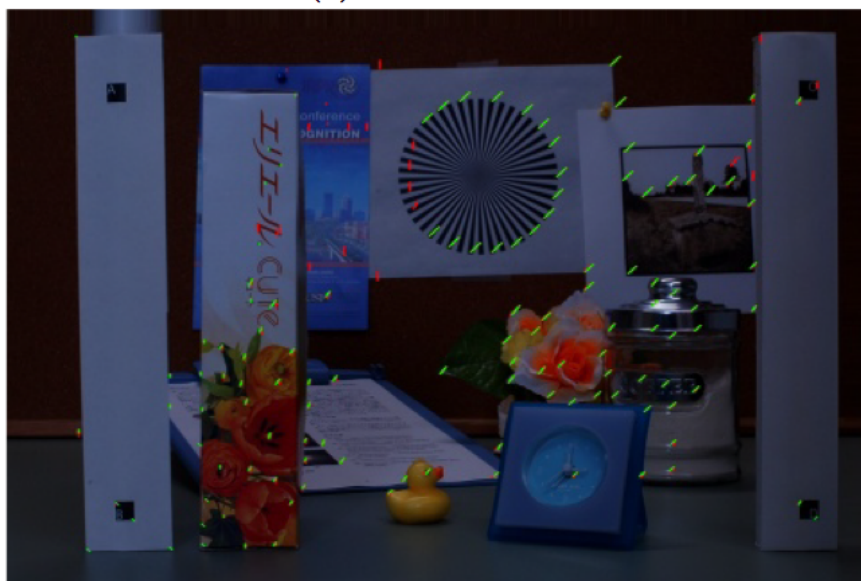
小型デジタルカメラで手持ち撮影した  
多視点撮影画像から大きなぼけを含む画像を得た



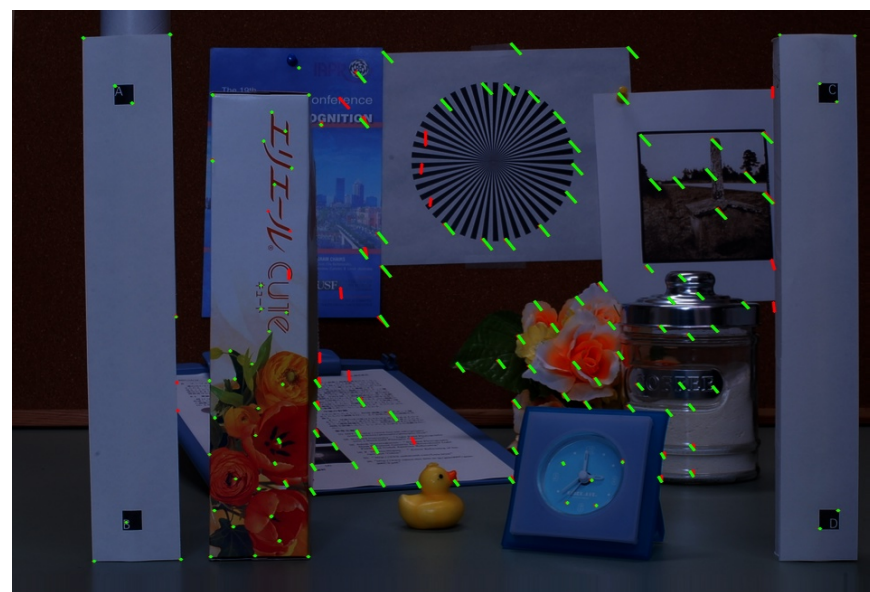
# エラーの検出



(a) KLT 追跡のみ



(b) 幾何的拘束を付加













KLT feature point tracking



Edge feature tracking 1

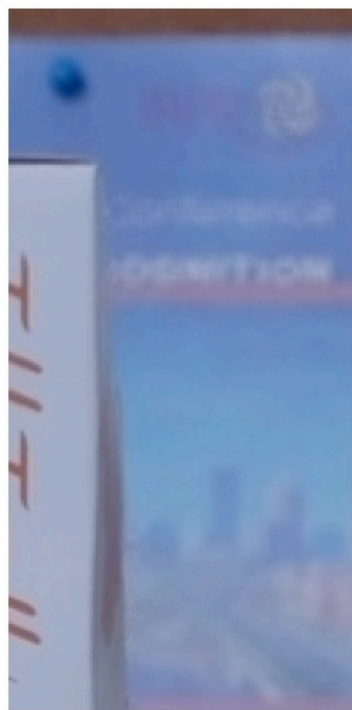
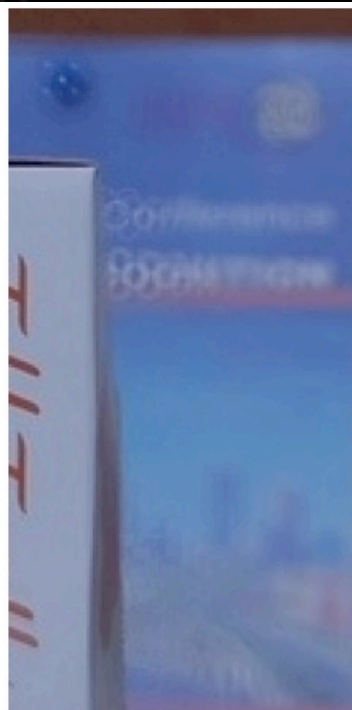


Edge feature tracking 2



Edge feature tracking 3





previous method (2008)



new method (2009)

