

# 超解像度技術を用いた 画像鮮明化システムの開発

Advanced Algorithm & Systems

篠原 修二

# 超解像とは



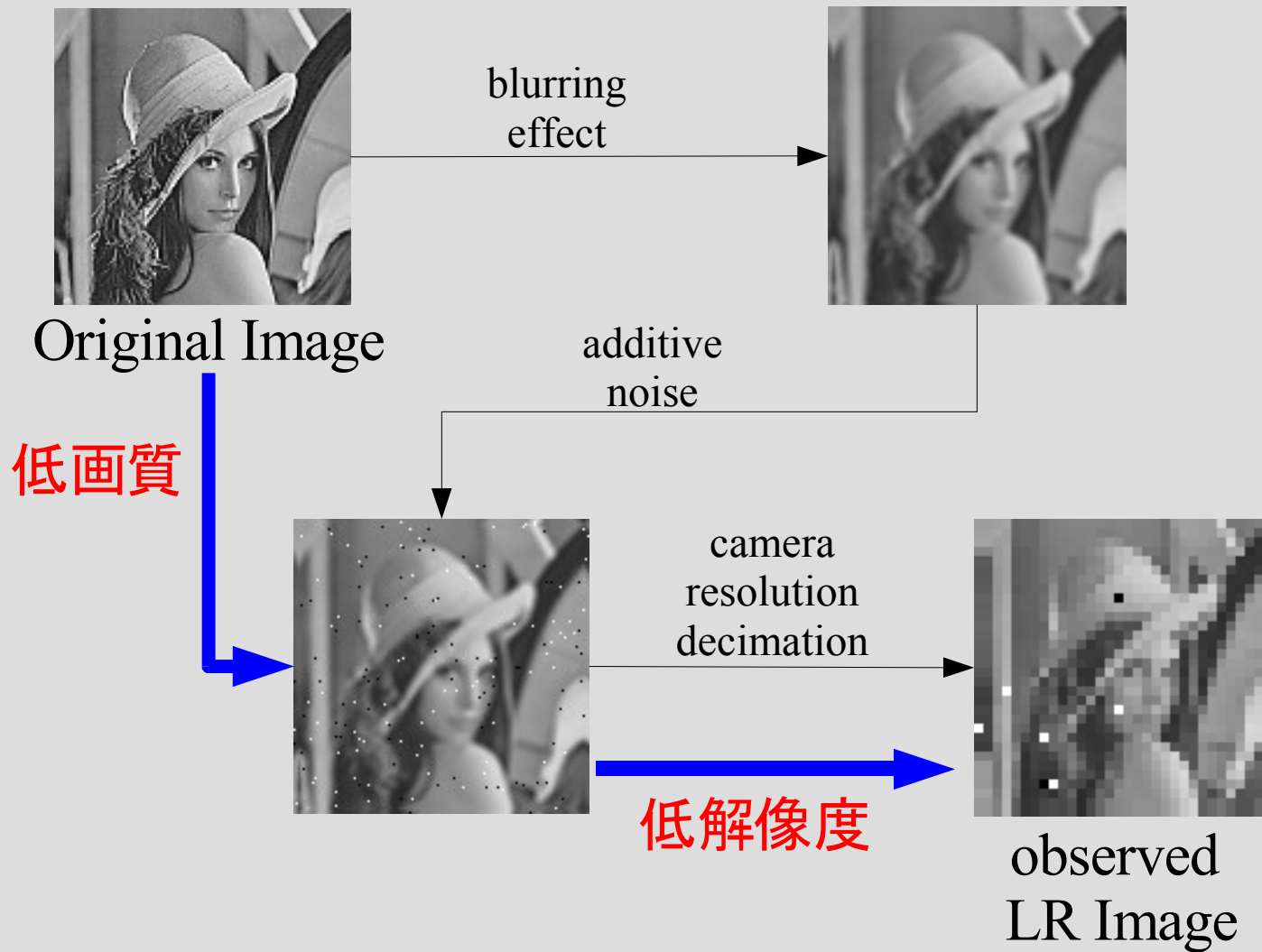
Low-Resolution(LR) images



High-Resolution(HR) image

⇒ ビデオなどから得られた複数枚の低解像度画像を用いて、一枚の鮮明な高解像度画像を作成すること

# 画質劣化の要因



# 超解像における二つの処理 1/2

## 1. Image Registration → 高解像度化

⇒ 変形 / 移動する被写体を捉えた複数枚の画像を重ね合わせることで、1枚の高解像度画像を作成



# 超解像における二つの処理 2/2

## 2. Deconvolution → 高画質化

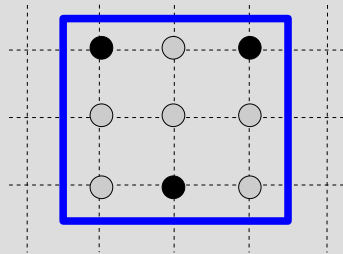
⇒ ピンぼけ、手ぶれ、被写体ぶれ、ノイズなどによって生じた画質の劣化を復元



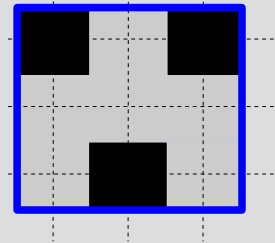
# Image Registration 1/3

## 基本的な考え方

時刻  $t$

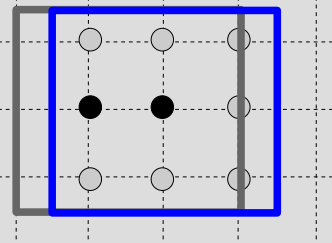


Low-Resolution Image( $I_1$ )



例：カメラが 0.5pixel 右に移動

時刻  $t+\Delta t$



Low-Resolution Image( $I_2$ )

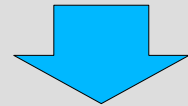
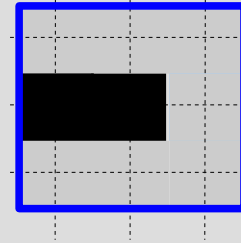
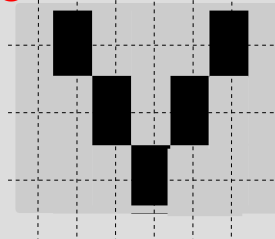
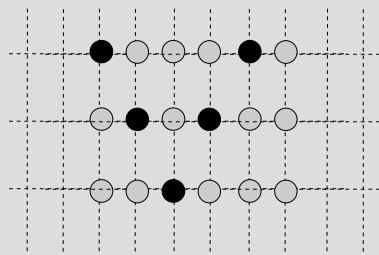


Image Registration( $I_1+I_2$ )

High-Resolution Image(横方向に  $\times 2$  の解像度)



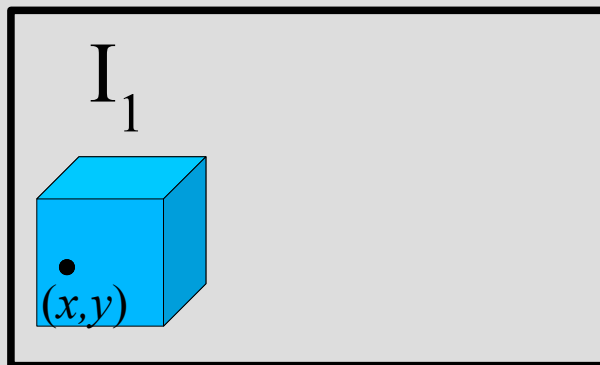
- データポイント
- Camera

# Image Registration 2/3

## Optical Flow Estimation

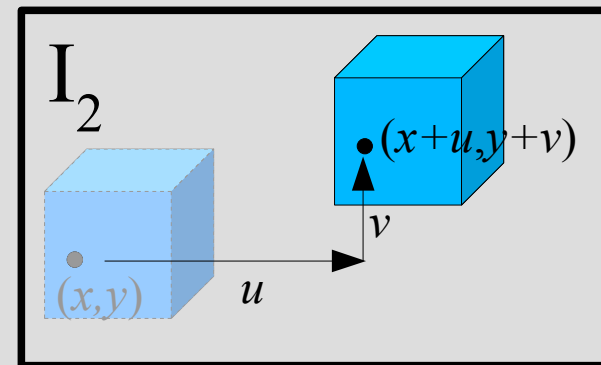
Object の移動  $(u, v)$  を sub-pixel レベルで推定

時刻  $t$



Translation

時刻  $t + \Delta t$



- Lucas-Kanade(LK) 法 (広く使われる Optical Flow Method)

constant-brightness assumption の下で

$$Err(u, v) = \sum_{x, y} (I_2(x+u, y+v) - I_1(x, y))^2$$

を最小化する  $\begin{pmatrix} u \\ v \end{pmatrix}$  を計算

# Image Registration 3/3

## ● Lucas-Kanade 法 (続き)

→ 具体的には

$$\begin{pmatrix} u \\ v \end{pmatrix} = A^{-1} b \quad \text{ただし,} \quad \begin{aligned} A &= \sum_{x,y} \nabla I(x,y) \nabla I(x,y)^T \\ b &= - \sum_{x,y} \nabla I(x,y) (I_2(x,y) - I_1(x,y)) \end{aligned}$$

→ Iterative の場合

$$b^{(i+1)} = - \sum_{x,y} \nabla I(x,y) I_2(x,y) + \sum_{x,y} \nabla I(x,y) I_1^{(i)}(x,y) \quad \text{とし,}$$

$I_1^{(i)}$  を逐次更新.

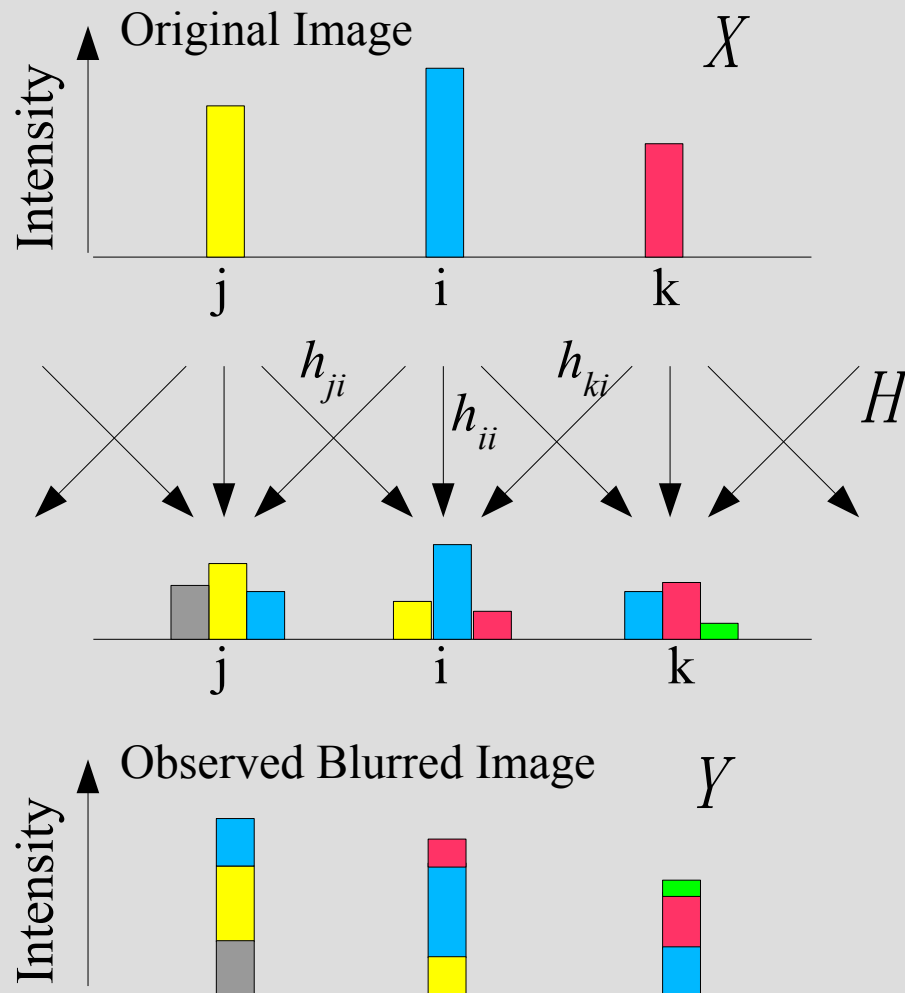
→ 大きい移動を伴う場合

Gaussian Pyramid を用いる



# Deconvolution 1/6

## 基本的な考え方



## Convolution

$$Y = H * X$$

$Y$  :blurred image(**observed**)

$X$  original image (**unknown**)

$H$  :filter (**known or unknown**)

$$y_i = \sum_j h_{ji} x_j \quad \text{ただし,}$$

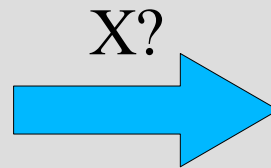
$$\sum_j h_{ji} = \sum_i h_{ji} = 1$$

# Deconvolution 2/6

## 基本的な考え方（続き）

### 逆問題としての Deconvolution

$$Y = H * X$$



$$\hat{X} = \operatorname{argmin}_X J(X)$$
$$J(X) = \|HX - Y\|_p^p$$

#### Deconvolution

$H$  が既知の場合,  $H$  と観測値  $Y$  から  $X$  を推定

#### Blind Deconvolution

$H$  が未知の場合, 観測値  $Y$  のみから  $H$  と  $X$  を推定

# Deconvolution 3/6

## 従来広く使われてきた手法 (1)

- Lucy-Richardson(LR) 法 (ベイズの定理を利用)

入力  $X$  を  $P(X)$ , 出力  $Y$  を  $P(Y)$ , フィルタ  $H$  を  $P(Y|X)$  とみなす

$P(X)$  が不明なため適当に  $P^t(X)$  を作成

Convolution により  $P^t(Y)$  を計算  $P^t(y_i) = \sum_k P^t(x_k) P(y_i|x_k)$

ベイズの定理を使って  $P(X|Y)$  を推定  $P^t(x_j|y_i) = P^t(x_j) P(y_i|x_j) / P^t(y_i)$

$P^t(X|Y)$  と  $P(Y)$  から新たな  $P^{t+1}(X)$  を作成  $P^{t+1}(x_j) = \sum_i P^t(x_j|y_i) P(y_i)$

まとめると

$$x_j \leftarrow x_j \sum_i \left( \frac{h_{ij} y_i}{\sum_k h_{ki} x_k} \right)$$

# Deconvolution 4/6

## 従来広く使われてきた手法 (2)

- Iterative Back Projection (IBP) 法

$$x_j \leftarrow x_j + \alpha \sum_i (y_i - \sum_k h_{kj} x_k) h_{ij}$$

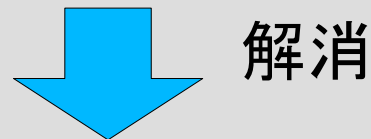
LR 法  $x_j \leftarrow x_j \sum_i \left( \frac{h_{ij} y_i}{\sum_k h_{ki} x_k} \right)$  と比較すると、

推定値  $x$  に基づいた Convolution の結果  $\sum_k h_{ki} x_k$  と観測値  $y$  の差を 0 としようとする (IBP 法) か、比を 1 にしようとするか (LR 法) の違いがある。

# Deconvolution 5/6

## 従来手法の問題点

- ノイズも忠実に再現してしまう
- Ringing Effect が見られる
- 処理速度が遅い
- フィルタ  $H$ (Point Spread Function) を正確に与える必要がある



Total Variation Method with Bilateral Filter (BTV 法)

# Deconvolution 6/6

- Bilateral Filter を用いた Total Variation(BTV) 法 **Fast and Robust**  
Regularization のための項を追加

$$J(X) = \|HX - Y\|_p^p + \lambda \rho(X)$$

TV 法では  $\rho(X) = \|\nabla X\|_1$  を用いる

↓ BTV 法

$$\hat{X} = \operatorname{argmin}_X [\|A(HX - Y)\|_1 + \lambda \underbrace{\sum_{l=-P}^P \sum_{m=0}^P}_{l+m \geq 0} \alpha^{|m|+|l|} \|X - S_x^l S_y^m X\|_1]$$

※ ノルムにマンハッタン距離を使う

※ Bilateral Filter を使う

$Y$  Image Registration 後の blurred HR image

$A$   $Y$  に値が存在する時にのみ計算を行うようにするための対角行列

$S_x^l, S_y^m$  水平方向に  $l$ , 垂直方向に  $m$  だけ  $X$  をシフトさせる Operator

**BTV 法では、ノイズ除去、補間、Deblurring を同時に行える**

# 処理概要

## 目的

ビデオシーケンスから取得した複数の低解像度（低画質）画像から1枚の高解像度（高画質）画像を作成すること

ビデオシーケンスから複数の静止画像を取得

Image Registration（高解像度化）

画像間の幾何変換計算に Gaussian Pyramid を用いた **LK 法** を使用

Deconvolution（高画質化）

原画像の推定（Deblurring），ノイズ除去，補間に **BTV 法** を使用

# Image Registration 評価 1/3

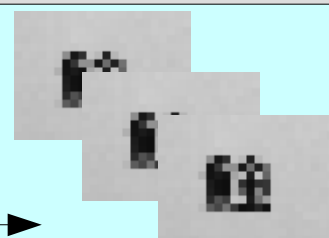
## 評価用入力データ

### 入力データ

#### 文字



1/4×1/4 Down Sampling(DS)



16 DS Images

+1% noise



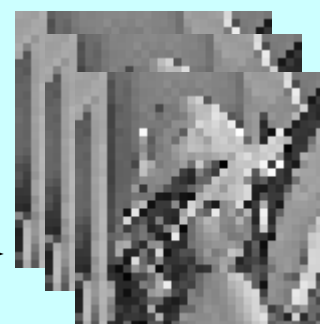
16 DS Images+noise

#### 顔



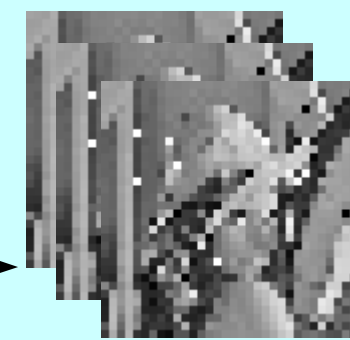
Original Image

1/4×1/4 Down Sampling(DS)



16 DS Images

+1% noise



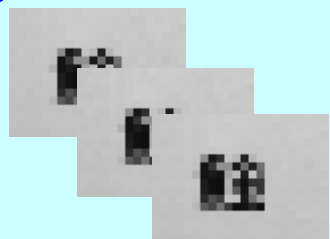
16 DS Images+noise



# Image Registration 評価 2/3

文字

入力



16 DS Images

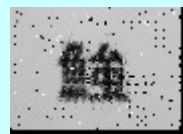
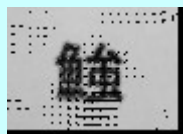


16 DS Images+noise

Image Registration  
(LK)

Interpolation  
(B-Spline)

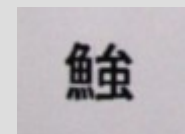
出力



+noise



+noise



Original Image

# Image Registration 評価 3/3

顔

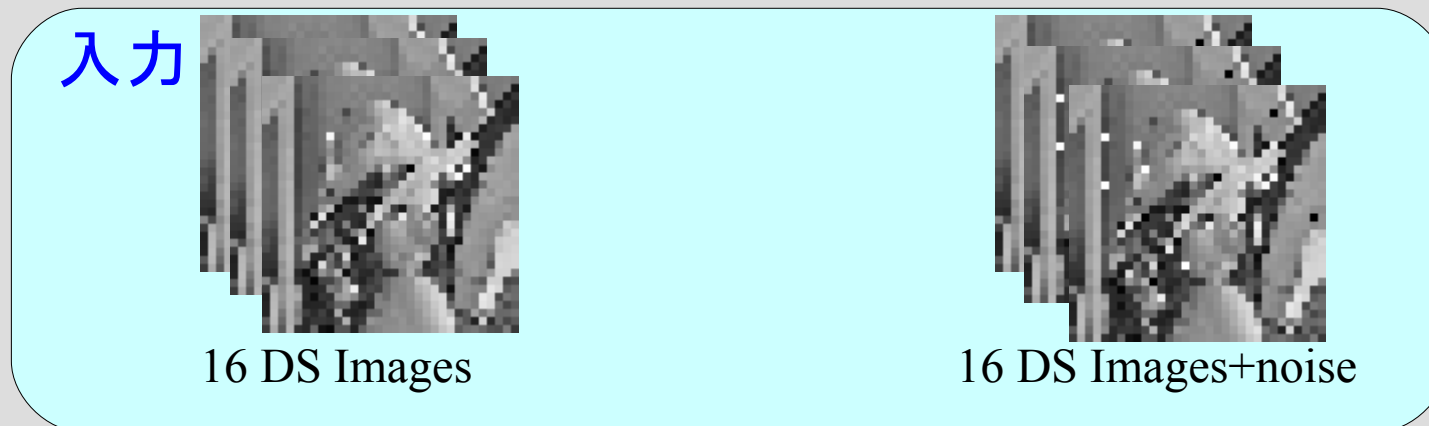
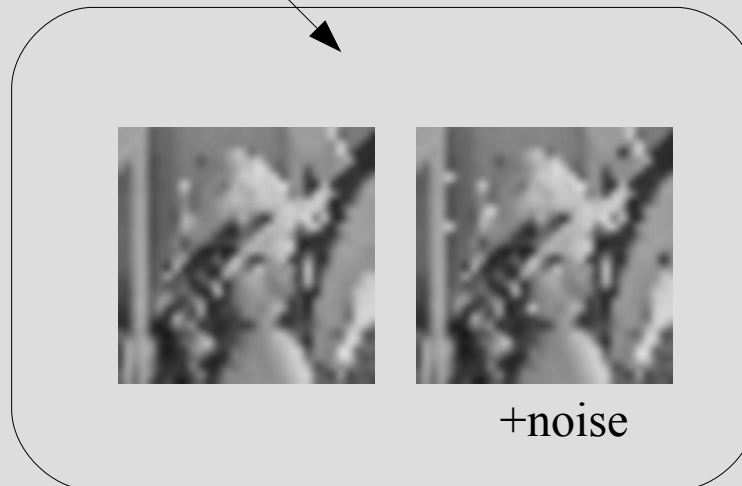
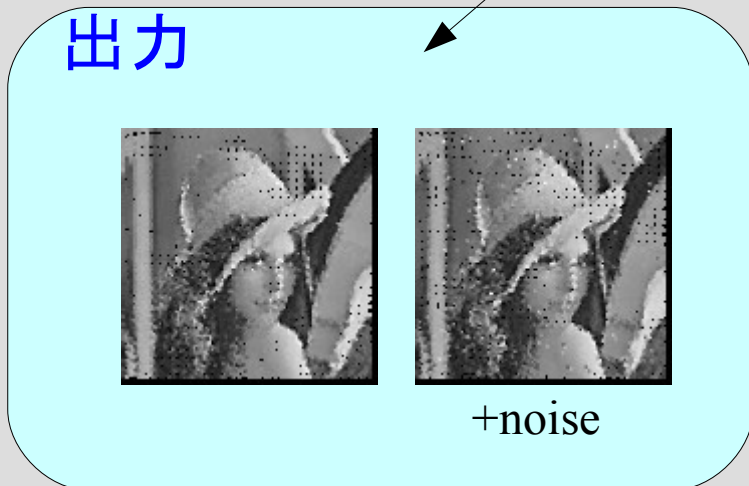


Image Registration  
(LK)

Interpolation  
(B-Spline)



# Deconvolution 評価 1/3

## 評価用入力データ

### 入力データ

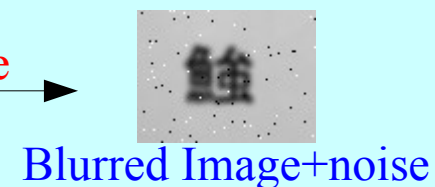
#### 文字



+blur( $\sigma=1.5$ )



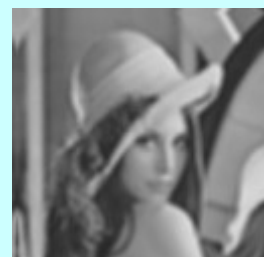
+1% noise



#### 顔



+blur( $\sigma=1.5$ )



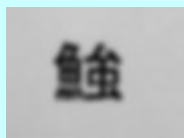
+1% noise



# Deconvolution 評価 2/3

文字

入力



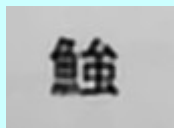
Blurred Image



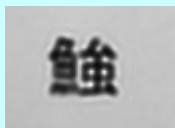
Blurred Image+noise

Deconvolution

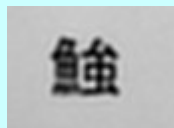
出力



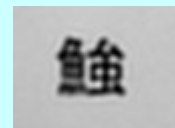
BTV



LR



IBP



BTV

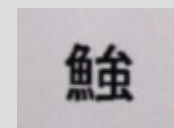


LR



IBP

+noise

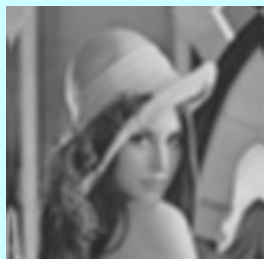


Original Image

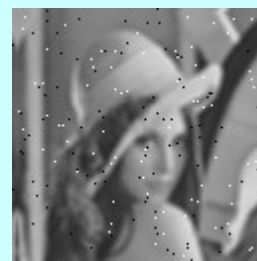
# Deconvolution 評価 3/3

顔

入力



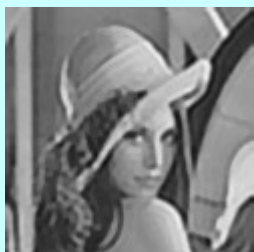
Blurred Image



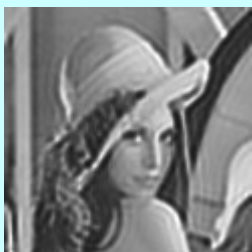
Blurred Image+noise

Deconvolution

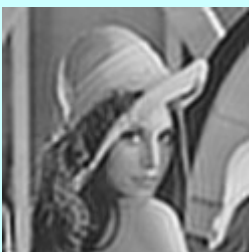
出力



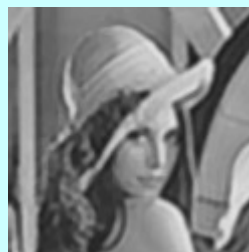
BTV



LR



IBP



BTV



LR



IBP

+noise



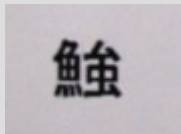
Original Image

# Image Registration+Deconvolution 複合評価 1/3

## 評価用入力データ

## 入力データ

### 文字



Original Image

blur( $\sigma=1.5$ ) +  
1/4×1/4 Down Sampling(DS)+  
1% noise



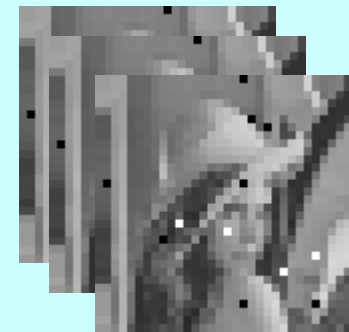
Blurred DS Image  
+noise

### 顔



Original Image

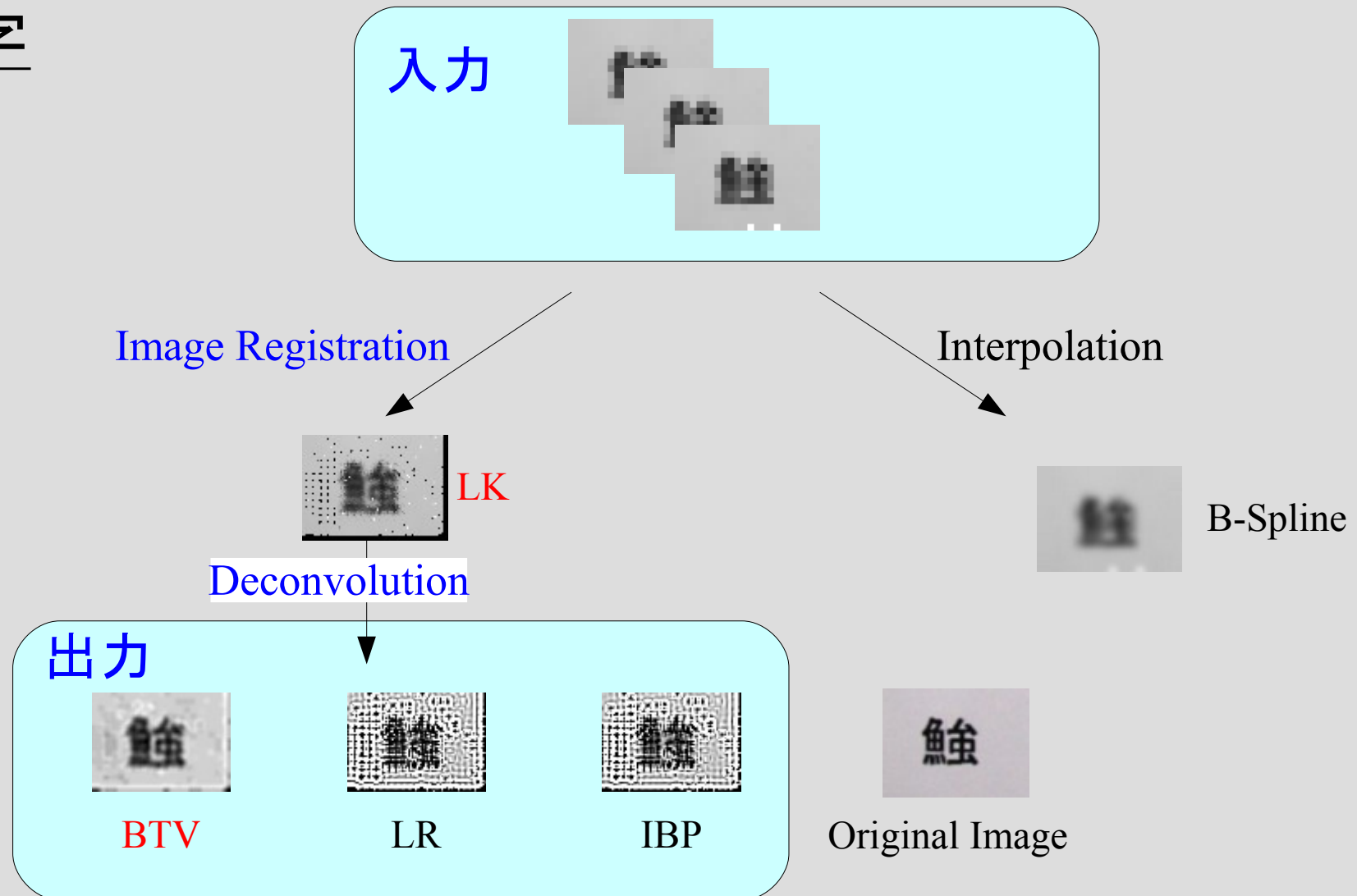
blur( $\sigma=1.5$ ) +  
1/4×1/4 Down Sampling(DS)+  
1% noise



Blurred DS Image  
+noise

# Image Registration+Deconvolution 複合評価 2/3

文字



# Image Registration+Deconvolution 評価 3/3

顔

入力

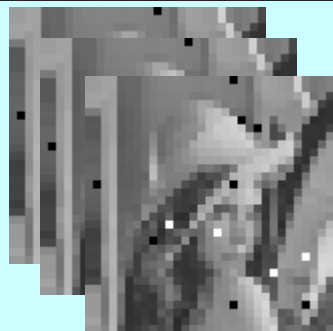
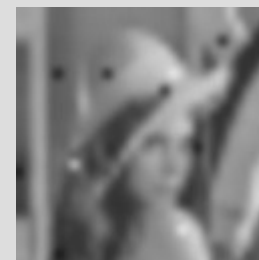


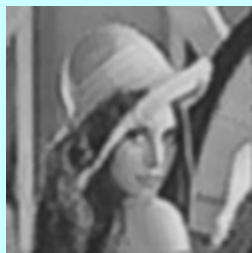
Image Registration(LK) +  
Deconvolution

Interpolation



B-Spline

出力



LK+BTV



LK+LR



LK+IBP



Original Image



# 実データでの評価

顔

入力



ビデオから切り出した  
10フレーム分画像

Zoom(8×8)



Image Registration(LK) +  
Deconvolution(BTV)

出力



Interpolation  
(B-Spline)



# 考察と課題 1/2

## ●Image Registration

- 小さい範囲での移動ならば、LK 法で sub-pixel レベルでの正確な移動推定が可能。
- Gaussian Pyramid を用いても 5pixel 以上の大きな移動に対しては精度が悪くなる（対策が必要）。
- Constant-brightness assumption が成立しない場合に対策が必要。
- Translation 以外の変換（affine 変換等）にも対応必要。

# 考察と課題 2/2

## ● Deconvolution

- BTV 法は, フィルタが既知でノイズがない場合には, LR 法や IBP 法と同等の性能を発揮する.
- BTV 法は, LR 法や IBP 法と比較してノイズに対して Robust である.
- Image Registration 処理を行った画像に対しては性能が落ちる.
- BTV 法は LR 法や IBP 法と同様にフィルタが既知の場合にしか使えない ⇒ Blind Deconvolution (PSF の自動推定)

## ● その他

- カラーへの対応.