# Linear Models
for
# Multi-Frame Super-Resolution Restoration
under
# Non-Affine Registration
and
# Spatially Varying PSF

Sean Borman & Robert L. Stevenson

Laboratory for Image and Signal Analysis

Department of Electrical Engineering

University of Notre Dame

Indiana, USA

# Introduction

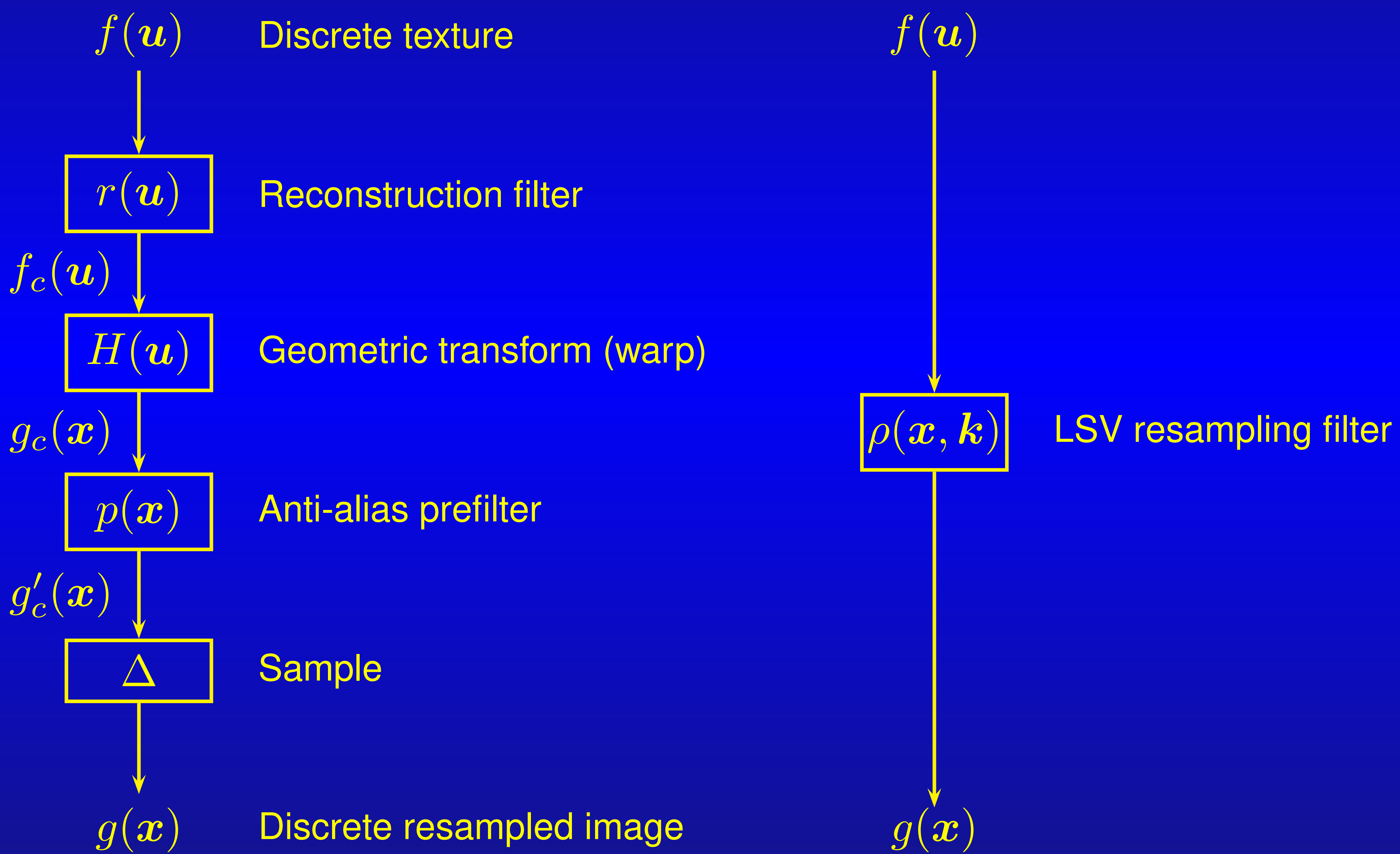
- Multi-frame super-resolution (SR) restoration
  - Image restoration from an image sequence
  - Exceeds resolving ability of classical single-frame methods
- Objectives
  - Generalize linear multi-frame observation model to . . .
    1. Non-affine image registration (e.g. projectivity)
    2. Spatially-varying PSF
  - Must be compatible with existing restoration framework
- Approach
  - Use result from image resampling/warping (computer graphics)
  - Propose algorithm for computing generalized observation model
  - Demonstrate applicaton to multi-frame super-resolution experiment

# Conceptual Image Resampling Pipeline (Heckbert)

$$
\begin{tikzcd}
f(\boldsymbol{u}) \arrow[d] \\
\boxed{r(\boldsymbol{u})} \arrow[d, "f_c(\boldsymbol{u})"'] \\
\boxed{H(\boldsymbol{u})} \arrow[d, "g_c(\boldsymbol{x})"'] \\
\boxed{p(\boldsymbol{x})} \arrow[d, "g'_c(\boldsymbol{x})"'] \\
\boxed{\triangle} \arrow[d] \\
g(\boldsymbol{x})
\end{tikzcd}
$$

| | | |
|---|---|---|
| $f(\boldsymbol{u})$ | Discrete texture | $\boldsymbol{u} = [u\ v]^T \in \mathbb{Z}^2$ |
| $r(\boldsymbol{u})$ | Reconstruction filter | |
| $H(\boldsymbol{u})$ | Geometric transform (warp) | $H : \boldsymbol{u} \longmapsto \boldsymbol{x}$ |
| $p(\boldsymbol{x})$ | Anti-alias prefilter | |
| $\triangle$ | Sample | |
| $g(\boldsymbol{x})$ | Discrete resampled image | $\boldsymbol{x} = [x\ y]^T \in \mathbb{Z}^2$ |

# Realizable Image Resampling Pipeline (Heckbert)



# LSV Image Resampling Filter

- Compute warped image from texture using

$$g(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^2} f(\boldsymbol{k}) \cdot \rho(\boldsymbol{x}, \boldsymbol{k}) \quad \text{for} \quad \boldsymbol{x}, \boldsymbol{k} \in \mathbb{Z}^2$$

- $\rho(\boldsymbol{x}, \boldsymbol{k})$ is a discrete linear, spatially varying (LSV) resampling filter

$$\rho(\boldsymbol{x}, \boldsymbol{k}) = \int p(\boldsymbol{x} - H(\boldsymbol{u})) \cdot r\,(\boldsymbol{u} - \boldsymbol{k}) \left| \frac{\partial H}{\partial \boldsymbol{u}} \right| \mathrm{d}\boldsymbol{u}$$

- The LSV resampling filter . . .
  - depends on the warp $H$
  - depends on the reconstruction filter $r$
  - is expressed in terms of a *warped* prefilter $p$
  - involves integration in texture space ($\boldsymbol{u}$)

# Multi-Frame Observation Model

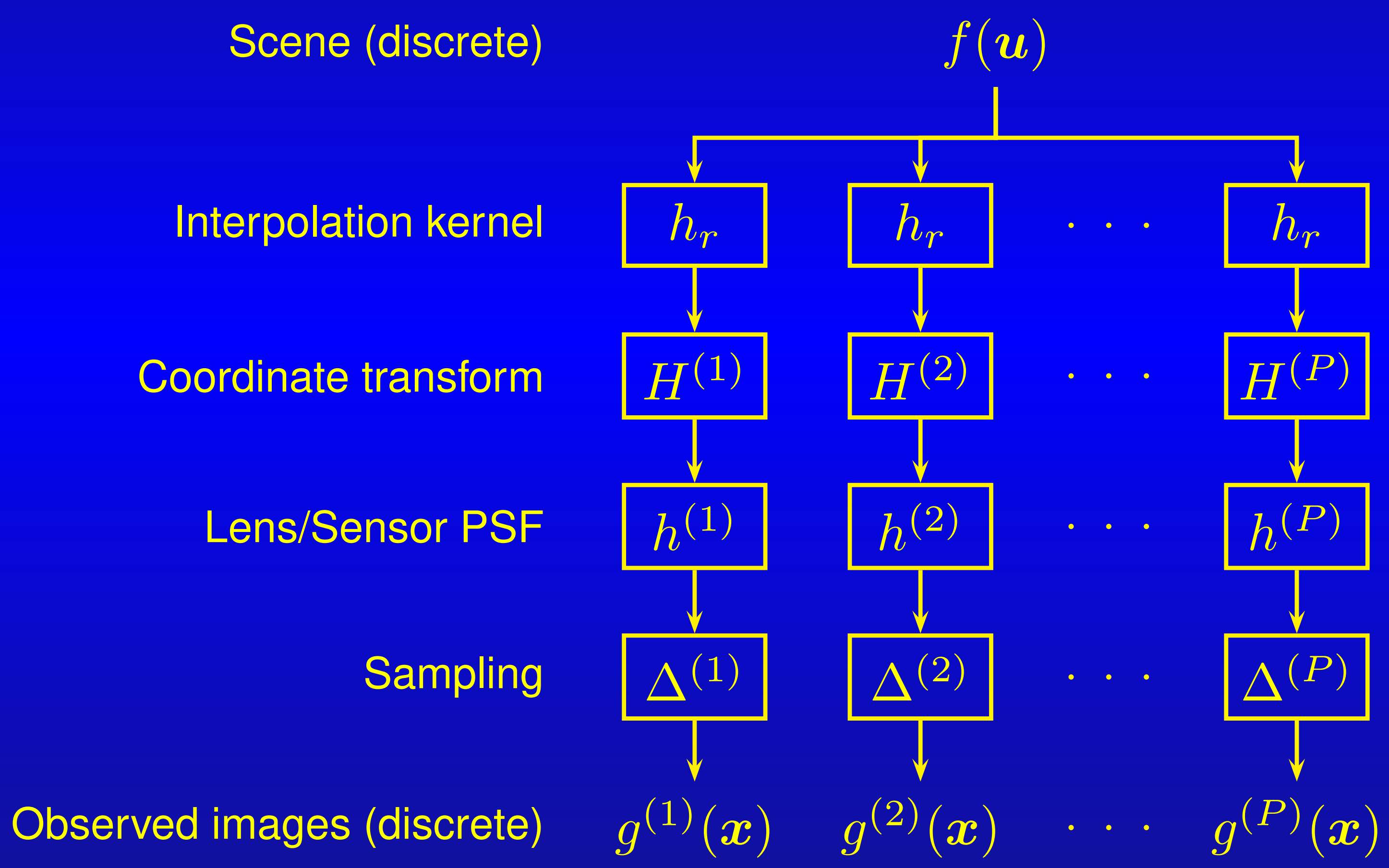
- Given images $g^{(i)}(\boldsymbol{x}),\ i \in \{1, 2, \ldots, P\}$ related to a continuous scene $f_c(\boldsymbol{u})$ via
  - coordinate transformations $H^{(i)} : \boldsymbol{u} \mapsto \boldsymbol{x}$ (scene/camera motion)
  - spatially varying PSF's $h^{(i)}$ (lens/sensor PSF, defocus, motion blur...)
  - spatial sampling
- Seek discretized approximation of $f_c(\boldsymbol{u})$ on high-resolution sampling lattice
- Using an interpolation kernel $h_r$ we approximate $f_c(\boldsymbol{u})$ as

$$f_c(\boldsymbol{u}) \approx \sum_{\boldsymbol{k} \in \mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot h_r\left(\boldsymbol{u} - \boldsymbol{V}\boldsymbol{k}\right) \quad \boldsymbol{k} \in \mathbb{Z}^2$$

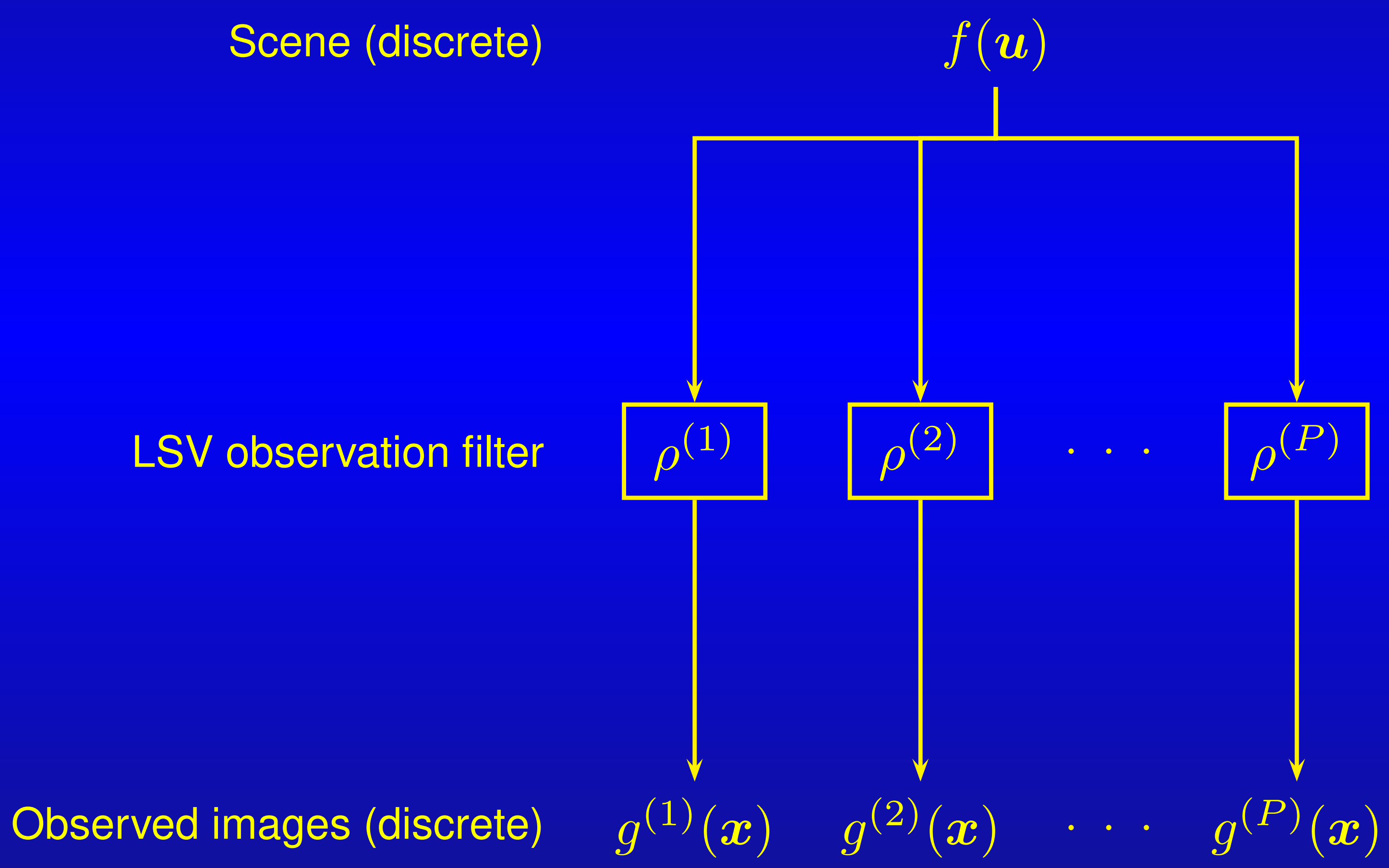
$\boldsymbol{V} = \begin{bmatrix} 1/Q_x & 0 \\ 0 & 1/Q_y \end{bmatrix}$ is a sampling matrix

$Q_x, Q_y \in \mathbb{N}$ are the horizontal and vertical magnification factors

# Discrete-Discrete Multi-Frame Observation Model



# Realizable Discrete-Discrete Multi-Frame Observation Model



# Realizable Discrete-Discrete Multi-Frame Observation Model

- We can relate $g^{(i)}(\boldsymbol{x})$ to $f(\boldsymbol{k})$ via LSV equations

$$g^{(i)}(\boldsymbol{x}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot \rho^{(i)}(\boldsymbol{x},\boldsymbol{k}) \quad \text{for} \quad \boldsymbol{x},\boldsymbol{k} \in \mathbb{Z}^2$$

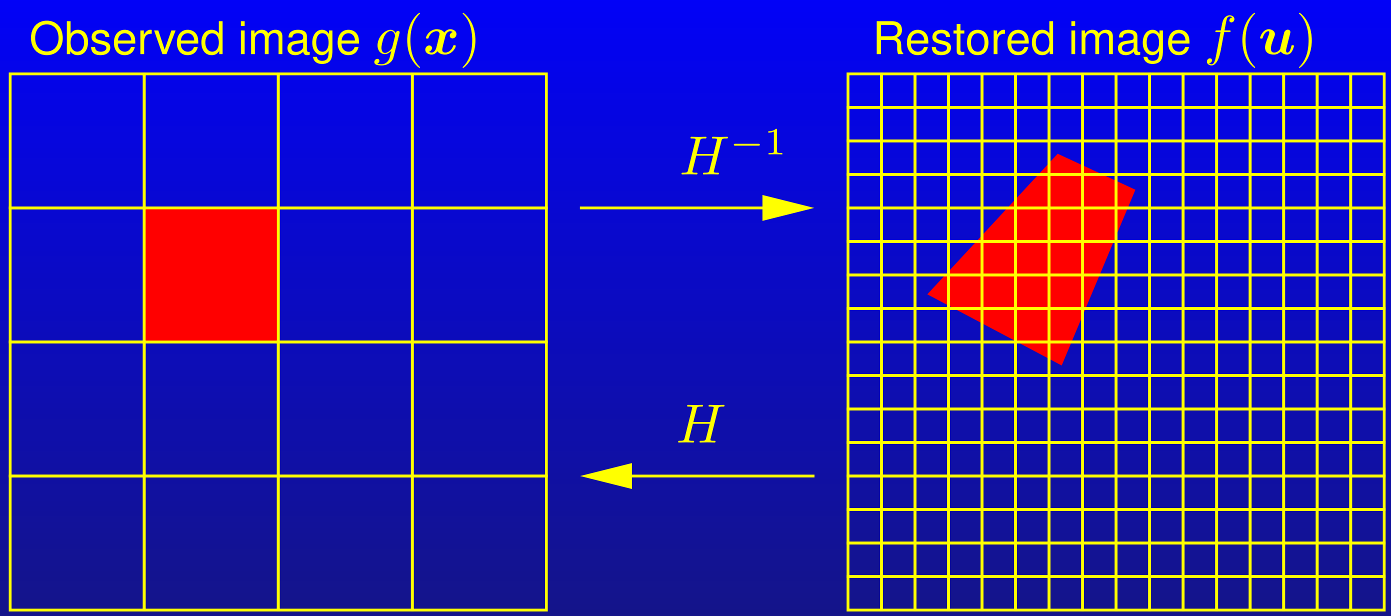
- $\rho^{(i)}(\boldsymbol{x},\boldsymbol{k})$ are discrete, linear spatially varying (LSV) observation filters

$$\rho^{(i)}(\boldsymbol{x},\boldsymbol{k}) = \int h^{(i)}\left(\boldsymbol{x}, H^{(i)}(\boldsymbol{u})\right) \cdot h_r\left(\boldsymbol{u} - \boldsymbol{V}\boldsymbol{k}\right) \left|\frac{\partial H^{(i)}}{\partial \boldsymbol{u}}\right| \mathrm{d}\boldsymbol{u}$$

- The LSV observation filters ...
  - depend on each coordinate transforms $H^{(i)}$
  - depend on the interpolation kernel $h_r$
  - are expressed in terms of the warped PSF's $h^{(i)}$
  - involve integration in the restoration space ($\boldsymbol{u}$)

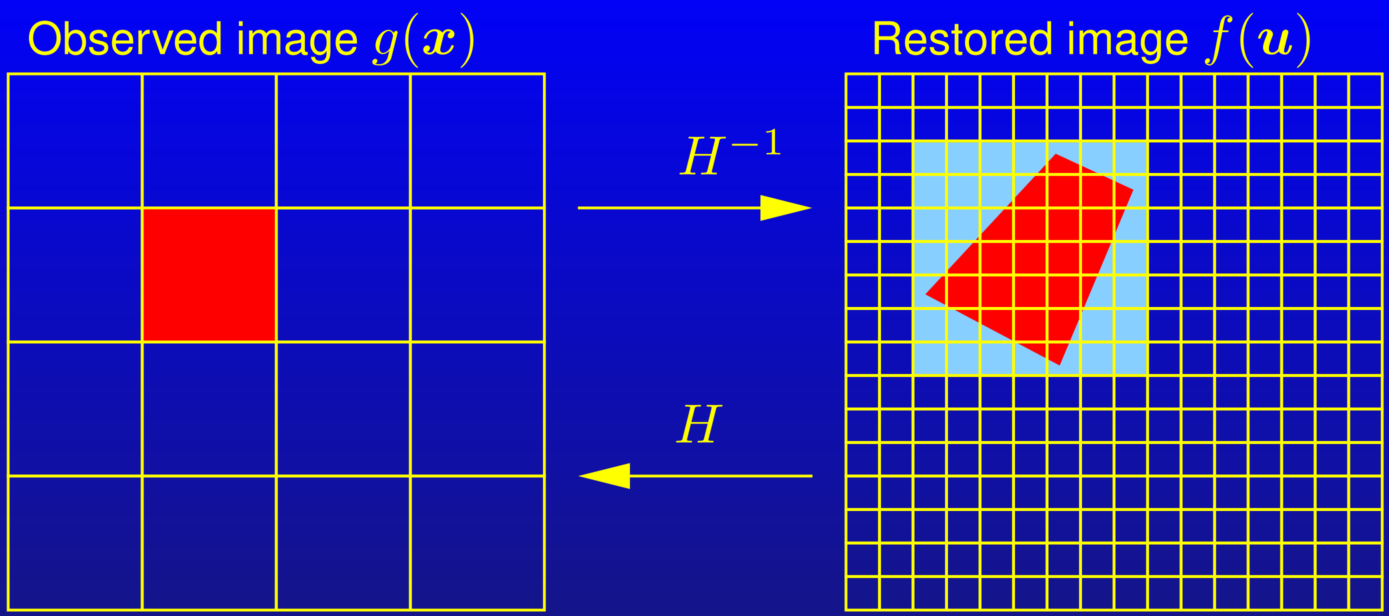
# Determining the Warped Pixel Response

1. Backproject PSF $h(\boldsymbol{x}, \boldsymbol{\alpha})$ from $g(\boldsymbol{x})$ to restored image using $H^{-1}$ (red)
2. 
3. 
4. 



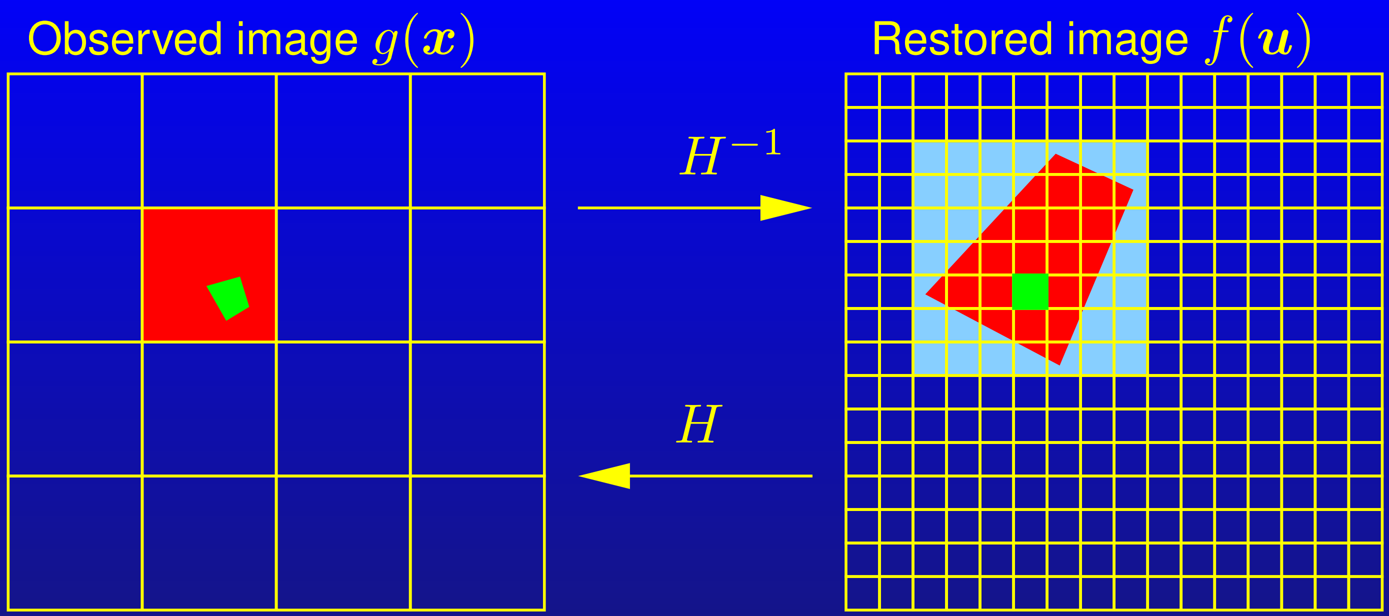
# Determining the Warped Pixel Response

1. Backproject PSF $h(\boldsymbol{x}, \boldsymbol{\alpha})$ from $g(\boldsymbol{x})$ to restored image using $H^{-1}$ (red)
2. Determine bounding region for image of $h(\boldsymbol{x}, \boldsymbol{\alpha})$ under backprojection (cyan)
3.
4.



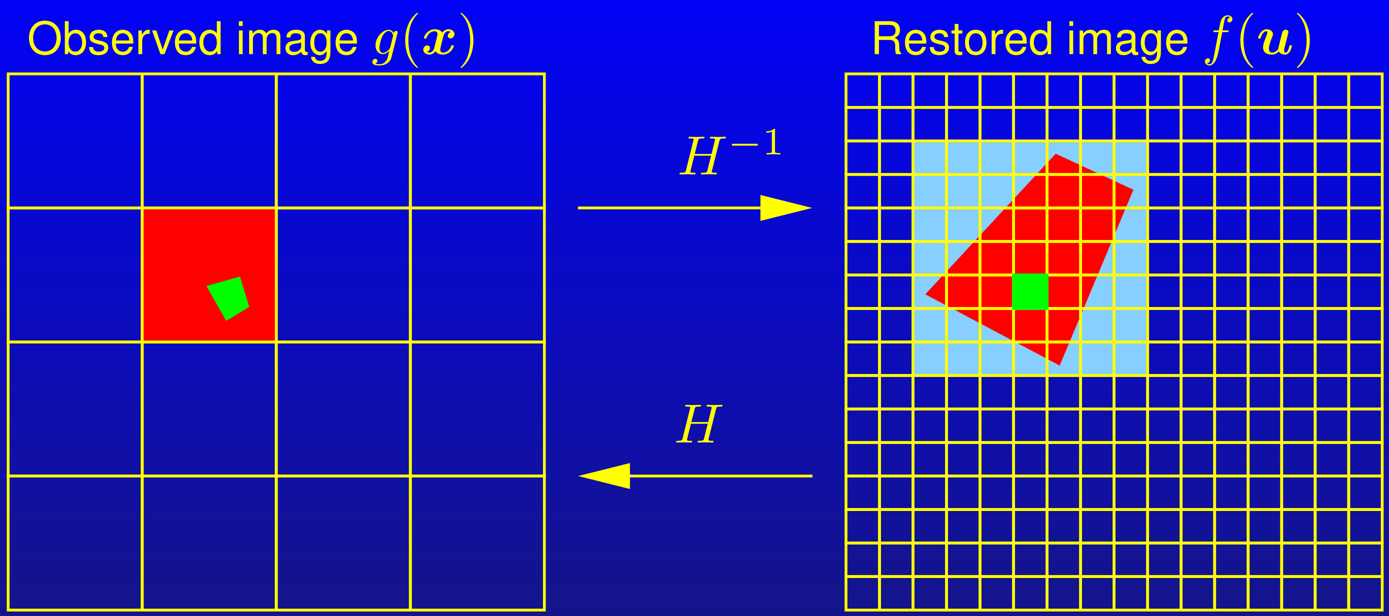
# Determining the Warped Pixel Response

1. Backproject PSF $h(\boldsymbol{x}, \boldsymbol{\alpha})$ from $g(\boldsymbol{x})$ to restored image using $H^{-1}$ (red)
2. Determine bounding region for image of $h(\boldsymbol{x}, \boldsymbol{\alpha})$ under backprojection (cyan)
3. $\forall$ S-R pixels $\boldsymbol{u}$ in region, project via $H$ and find $h(\boldsymbol{x}, H(\boldsymbol{u}))$ (green)
4. 



# Determining the Warped Pixel Response

1. Backproject PSF $h(\boldsymbol{x}, \boldsymbol{\alpha})$ from $g(\boldsymbol{x})$ to restored image using $H^{-1}$ (red)
2. Determine bounding region for image of $h(\boldsymbol{x}, \boldsymbol{\alpha})$ under backprojection (cyan)
3. $\forall$ S-R pixels $\boldsymbol{u}$ in region, project via $H$ and find $h(\boldsymbol{x}, H(\boldsymbol{u}))$ (green)
4. Scale according to Jacobian and interpolation kernel $h_r$ then integrate over $\boldsymbol{u}$



# Algorithm to Determine the Observation Filter

for each observed image $\boldsymbol{g}^{(i)}$ $\{$

for each pixel $\boldsymbol{x}$ $\{$

back-project the boundary of $h^{(i)}(\boldsymbol{x}, \boldsymbol{\alpha})$ from $g^{(i)}(\boldsymbol{x})$

to the restored image space using ${H^{(i)}}^{-1}$

determine a bounding region for the image of $h(\boldsymbol{x}, \boldsymbol{\alpha})$ under ${H^{(i)}}^{-1}$

for each pixel indexed by $\boldsymbol{k}$ in the bounding region $\{$

set $\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k}) = h^{(i)}\left(\boldsymbol{x}, H^{(i)}(\boldsymbol{u})\right) \cdot \left|\frac{\partial H^{(i)}}{\partial \boldsymbol{u}}\right|$ with $\boldsymbol{u} = \boldsymbol{V}\boldsymbol{k}$

$\}$

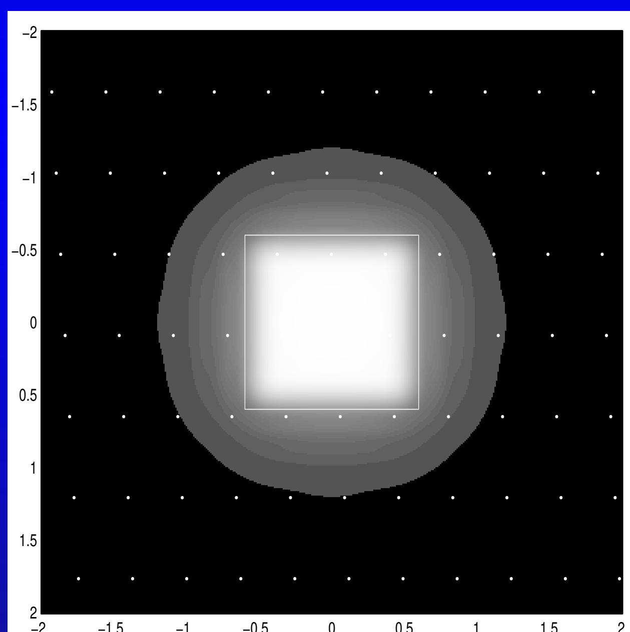
normalize $\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k})$ so that $\sum_k \rho^{(i)}(\boldsymbol{x}, \boldsymbol{k}) = 1$

$\}$

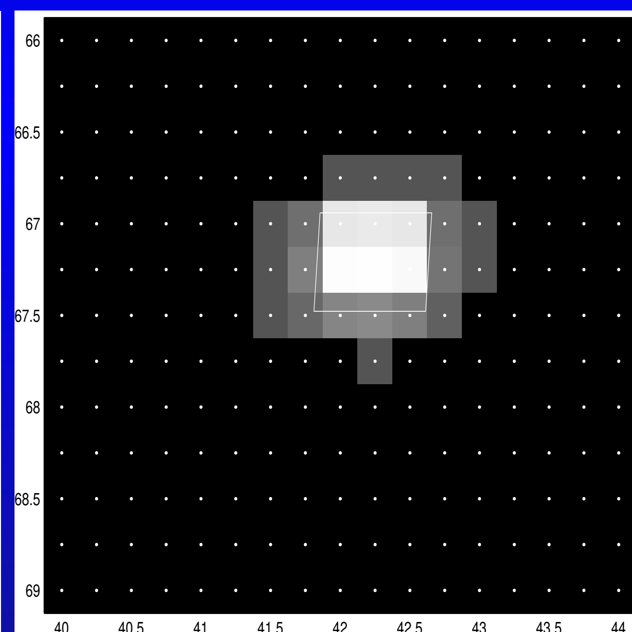
$\}$

# Example of Observation Filter

- 100% fill-factor pixel
- Diffraction-limited optics
- Projective spatial transformation
- $Q_x = Q_y = 4$




Observed Pixel PSF




Observation filter $\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k})$

# Matrix-Vector Linear Multi-Frame Observation Model

- Represent images as lexicographically ordered vectors $\boldsymbol{g}^{(i)}$, $\boldsymbol{f}$ (finite case)

  $\Rightarrow$ single pixel observation may then be written as an inner product.

$$g^{(i)}(\boldsymbol{x}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \rho^{(i)}(\boldsymbol{x},\boldsymbol{k}) \cdot f(\boldsymbol{V}\boldsymbol{k}) \qquad \text{or equivalently} \qquad \boldsymbol{g}_j^{(i)} = \left\langle A_j^{(i)}, \boldsymbol{f} \right\rangle$$

- Stack inner product equations to get single image matrix-vector equation

$$\boldsymbol{g}^{(i)} = \boldsymbol{A}^{(i)}\boldsymbol{f}$$

- Stack matrices $\boldsymbol{A}^{(i)}$ and observations $\boldsymbol{g}^{(i)}$ to get

$$\boldsymbol{g} \doteq \begin{bmatrix} \boldsymbol{g}^{(1)} \\ \boldsymbol{g}^{(2)} \\ \vdots \\ \boldsymbol{g}^{(P)} \end{bmatrix} \quad \text{and} \quad \boldsymbol{A} \doteq \begin{bmatrix} \boldsymbol{A}^{(1)} \\ \boldsymbol{A}^{(2)} \\ \vdots \\ \boldsymbol{A}^{(P)} \end{bmatrix} \quad \text{so that we have} \quad \boldsymbol{g} = \boldsymbol{A}\boldsymbol{f}$$

## A Bayesian Framework for Restoration

- Classic linear inverse problem
- Ill-posed, so use regularized solution method with *a-priori* information
- Use augmented observation model which includes noise

$$\boldsymbol{g} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{n}$$

- $\hat{\boldsymbol{f}}_{\mathrm{MAP}}$ maximizes the *a-posteriori* probability $\mathcal{P}\left(\boldsymbol{f}|\boldsymbol{g}\right)$

$$\begin{aligned}
\hat{\boldsymbol{f}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{f}} \left\{\mathcal{P}\left(\boldsymbol{f}|\boldsymbol{g}\right)\right\} \\
&= \arg\max_{\boldsymbol{f}} \left\{\frac{\mathcal{P}\left(\boldsymbol{g}|\boldsymbol{f}\right)\mathcal{P}\left(\boldsymbol{f}\right)}{\mathcal{P}\left(\boldsymbol{g}\right)}\right\} \\
&= \arg\max_{\boldsymbol{f}} \left\{\log\mathcal{P}\left(\boldsymbol{g}|\boldsymbol{f}\right) + \log\mathcal{P}\left(\boldsymbol{f}\right)\right\}
\end{aligned}$$

## A Bayesian Framework for Restoration

- Likelihood term: Assume noise is zero-mean Gaussian

$$\begin{aligned}\mathcal{P}\left(\boldsymbol{g}|\boldsymbol{f}\right) &= \mathcal{P}_{\boldsymbol{N}}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f}) \\ &\propto \exp\left\{-\frac{1}{2}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})^T\boldsymbol{K}^{-1}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})\right\}\end{aligned}$$

- Prior term: Markov random field (Gibbs density)

$$\mathcal{P}\left(\boldsymbol{f}\right) \propto \exp\left\{-\frac{1}{\beta}\sum_{c\in\mathcal{C}}\rho_T\left(\boldsymbol{\partial}_c\boldsymbol{f}\right)\right\}$$

  - Huber penalty function $\rho_T(x)$ (edge preserving)
  - Local interactions $\boldsymbol{\partial}_c$ approximate $2^{\text{nd}}$ spatial derivates in 4 orientations

## A Bayesian Framework for Restoration

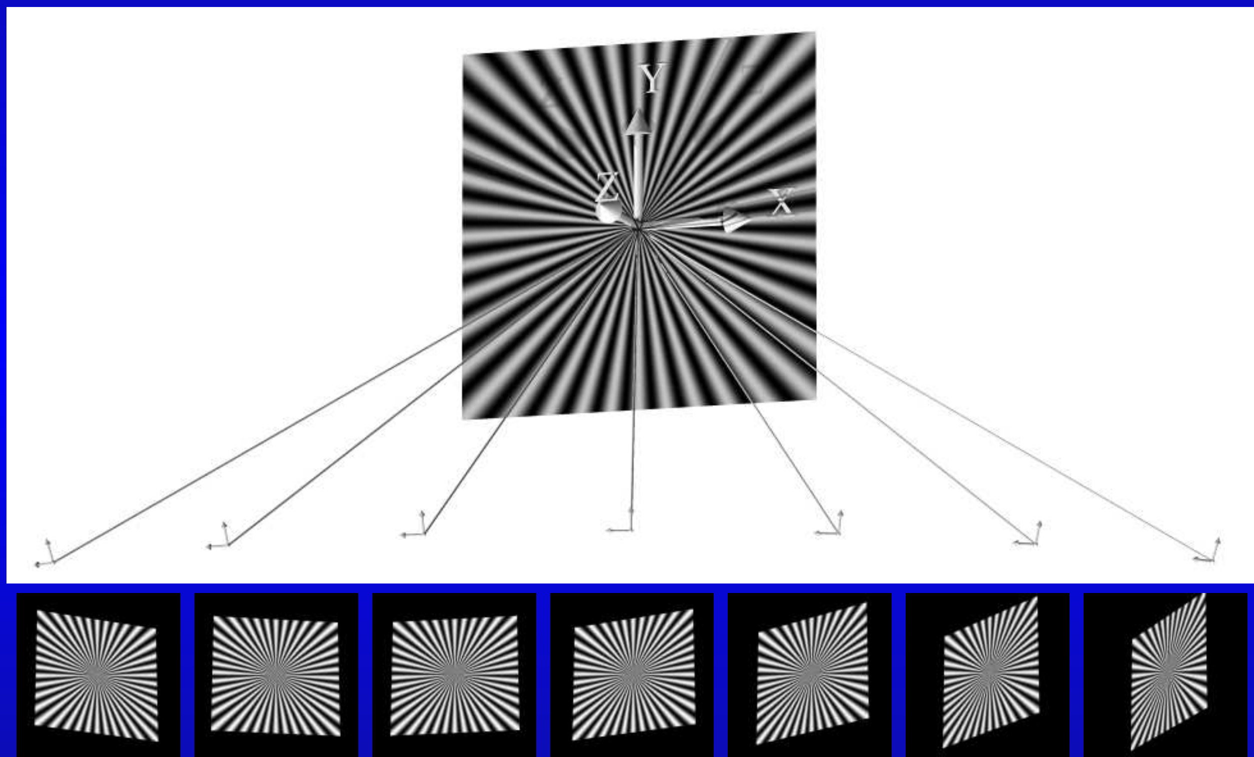
- Combined objective function

$$\begin{aligned}\hat{\boldsymbol{f}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{f}} \left\{ -\frac{1}{2}(\boldsymbol{g} - \boldsymbol{A}\boldsymbol{f})^T \boldsymbol{K}^{-1} (\boldsymbol{g} - \boldsymbol{A}\boldsymbol{f}) - \frac{1}{\beta} \sum_{c \in \mathcal{C}} \rho_T \left( \boldsymbol{\partial}_c \boldsymbol{f} \right) \right\} \\ &= \arg\min_{\boldsymbol{f}} \left\{ \frac{1}{2}(\boldsymbol{g} - \boldsymbol{A}\boldsymbol{f})^T \boldsymbol{K}^{-1} (\boldsymbol{g} - \boldsymbol{A}\boldsymbol{f}) + \gamma \sum_{c \in \mathcal{C}} \rho_T \left( \boldsymbol{\partial}_c \boldsymbol{f} \right) \right\}\end{aligned}$$

- Use your favorite optimization technique to find $\hat{\boldsymbol{f}}_{\mathrm{MAP}}$
- Unique solution under very mild conditions

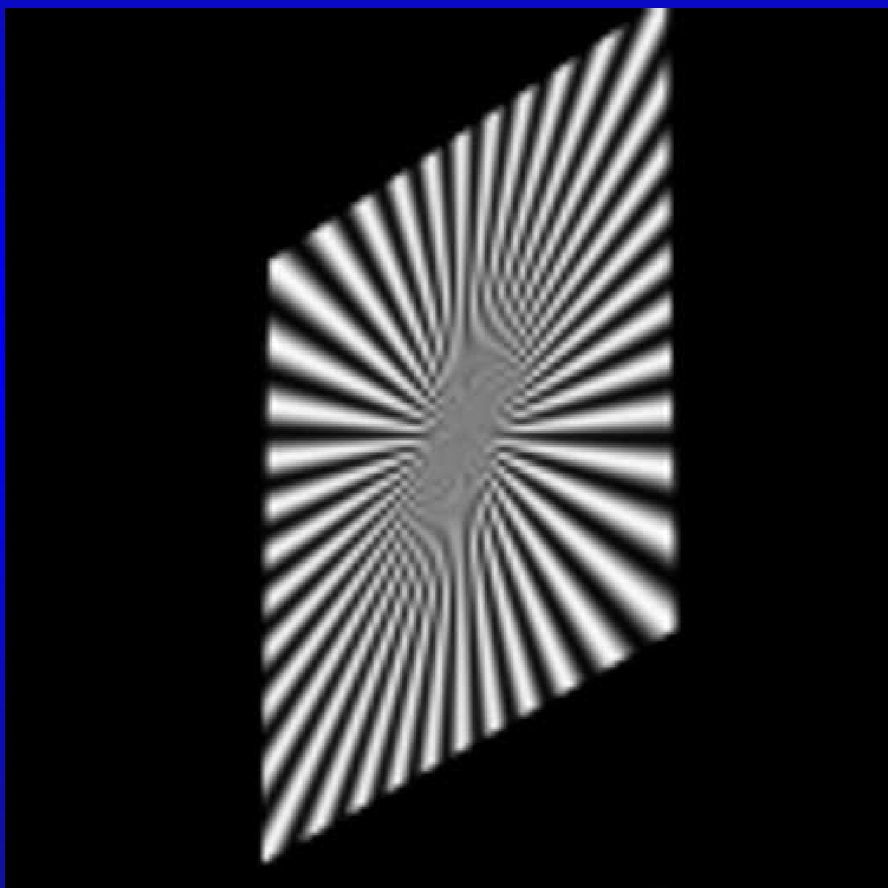
# Example

- Simulated imaging environment

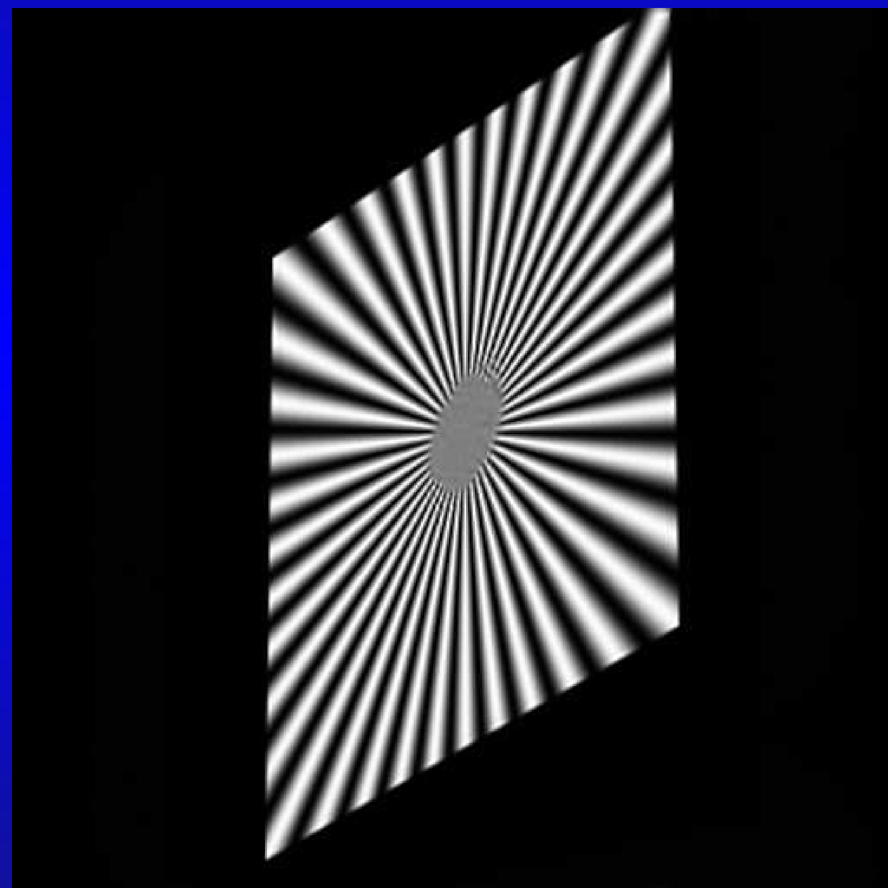


# Example

- Super-Resolution Restoration



Cubic spline interpolation



Multi-frame restoration

# Summary

- Generalized linear observation model used in multi-frame super-resolution restoration
  - Non-affine image registration
  - Easy to accommodate spatially-varying PSFs
- Algorithm to find linear, spatially varying observation filter
- Leads to sparse observation matrix (construct only once)
- Well-suited to iterative restoration methods
- No changes to restoration framework necessary
- Demonstrate application

end

# Image Resampling

- Objective: Sampling of discrete image under coordinate transformation
- Discrete input image (texture): $f(\boldsymbol{u})$ with $\boldsymbol{u} = [u\ v]^T \in \mathbb{Z}^2$
- Discrete output image (warped): $g(\boldsymbol{x})$ with $\boldsymbol{x} = [x\ y]^T \in \mathbb{Z}^2$
- Forward mapping: $H : \boldsymbol{u} \longmapsto \boldsymbol{x}$
- Simplistic approach: $\forall \boldsymbol{x} \in \mathbb{Z}^2, \quad g(\boldsymbol{x}) = f(H^{-1}(\boldsymbol{x}))$
- Problems:
  1. $H^{-1}(\boldsymbol{x})$ need not fall on sample points (interpolation required)
  2. $H^{-1}(\boldsymbol{x})$ may undersample $f(\boldsymbol{u})$ resulting in aliasing

     (This occurs when the the mapping results in minification)

# Conceptual Image Resampling Pipeline

1. Continuous reconstruction (interpolation) of input image (texture):

$$f_c(\boldsymbol{u}) = f(\boldsymbol{u}) \circledast r(\boldsymbol{u}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^2} f(\boldsymbol{k}) \cdot r(\boldsymbol{u} - \boldsymbol{k})$$

2. Warp the continuous reconstruction:

$$g_c(\boldsymbol{x}) = f_c\left(H^{-1}(\boldsymbol{x})\right)$$

3. Apply the anti-alias prefilter $p(\boldsymbol{x})$:

$$g_c'(\boldsymbol{x}) = g_c(\boldsymbol{x}) \circledast p(\boldsymbol{x}) = \int g_c(\boldsymbol{\alpha}) \cdot p(\boldsymbol{x} - \boldsymbol{\alpha}) \, \mathrm{d}\boldsymbol{\alpha}$$

4. Sample to produce the discrete output image:

$$g(\boldsymbol{x}) = g_c'(\boldsymbol{x}) \quad \text{for} \quad \boldsymbol{x} \in \mathbb{Z}^2$$

# Realizable Image Resampling Pipeline

- Never reconstruct continuous images:

$$
\begin{aligned}
g(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} &= g'_c(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} \\
&= \int f_c\left(H^{-1}(\boldsymbol{\alpha})\right) \cdot p(\boldsymbol{x}-\boldsymbol{\alpha})\,\mathrm{d}\boldsymbol{\alpha} \\
&= \int p(\boldsymbol{x}-\boldsymbol{\alpha}) \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{k}) \cdot r\left(H^{-1}(\boldsymbol{\alpha})-\boldsymbol{k}\right)\mathrm{d}\boldsymbol{\alpha} \\
&= \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{k})\rho(\boldsymbol{x},\boldsymbol{k})
\end{aligned}
$$

where

$$
\rho(\boldsymbol{x},\boldsymbol{k}) = \int p(\boldsymbol{x}-\boldsymbol{\alpha}) \cdot r\left(H^{-1}(\boldsymbol{\alpha})-\boldsymbol{k}\right)\mathrm{d}\boldsymbol{\alpha}
$$

is a <u>spatially varying</u> resampling filter.

# Realizable Image Resampling Pipeline

- Consider the resampling filter

$$\rho(\boldsymbol{x}, \boldsymbol{k}) = \int p(\boldsymbol{x} - \boldsymbol{\alpha}) \cdot r\left(H^{-1}(\boldsymbol{\alpha}) - \boldsymbol{k}\right) \mathrm{d}\boldsymbol{\alpha}$$

  - expressed i.t.o. warped reconstruction filter $r$
  - integration in $\boldsymbol{x}$-space (warped)

- Change variables $\boldsymbol{\alpha} = H(\boldsymbol{u})$

$$\rho(\boldsymbol{x}, \boldsymbol{k}) = \int p(\boldsymbol{x} - H(\boldsymbol{u})) \cdot r\left(\boldsymbol{u} - \boldsymbol{k}\right) \left|\frac{\partial H}{\partial \boldsymbol{u}}\right| \mathrm{d}\boldsymbol{u}$$

  - expressed i.t.o. the warped prefilter $p$
  - integrate in $\boldsymbol{u}$-space (texture)

# Multi-Frame Observation Model

- Observe low resolution image sequence $g^{(i)}(\boldsymbol{x}),\ i \in \{1, 2, \ldots, P\}$
- Observations derive from a continuous scene $f_c(\boldsymbol{u})$
- Related via:
  - Coordinate transformations $H^{(i)} : \boldsymbol{u} \mapsto \boldsymbol{x}$ (scene/camera motion)
  - Spatially varying PSF's $h^{(i)}$ (lens/sensor PSF, defocus, motion blur...)
  - Spatial sampling

$$g^{(i)}(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \int h^{(i)}(\boldsymbol{x}, \boldsymbol{\alpha}) \cdot f_c\left(H^{(i)^{-1}}(\boldsymbol{\alpha})\right) \mathrm{d}\boldsymbol{\alpha}.$$

# Discrete-Discrete Multi-Frame Observation Model

- Seek discretized approximation of $f_c(\boldsymbol{u})$ on high-resolution sampling lattice
- Interpolate samples $f(\boldsymbol{k}),\ \boldsymbol{k} \in \mathbb{Z}^2$ using kernel $h_r$

$$f_c(\boldsymbol{u}) \approx \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot h_r(\boldsymbol{u} - \boldsymbol{V}\boldsymbol{k})$$

  – $\boldsymbol{V}$ is the sampling matrix

$$\boldsymbol{V} = \begin{bmatrix} 1/Q_x & 0 \\ 0 & 1/Q_y \end{bmatrix}$$

  – $Q_x$ and $Q_y$ are horizontal and vertical sampling densities

# Discrete-Discrete Multi-Frame Observation Model

- Combine with earlier result:

$$g^{(i)}(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \int h^{(i)}(\boldsymbol{x},\boldsymbol{\alpha}) \cdot f_c\left(H^{(i)^{-1}}(\boldsymbol{\alpha})\right) \mathrm{d}\boldsymbol{\alpha}$$

$$= \int h^{(i)}(\boldsymbol{x},\boldsymbol{\alpha}) \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot h_r\left(H^{(i)^{-1}}(\boldsymbol{\alpha}) - \boldsymbol{V}\boldsymbol{k}\right) \mathrm{d}\boldsymbol{\alpha}$$

- Compare with resampling expression:

$$g(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \int p(\boldsymbol{x}-\boldsymbol{\alpha}) \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{k}) \cdot r\left(H^{-1}(\boldsymbol{\alpha}) - \boldsymbol{k}\right) \mathrm{d}\boldsymbol{\alpha}$$

- Identical in form to resampling expressions

# Discrete-Discrete Multi-Frame Observation Model

$\Rightarrow$ Define spatially varying observation filter (c.f. resampling filter)

$$\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k}) = \int h^{(i)}(\boldsymbol{x}, \boldsymbol{\alpha}) \cdot h_r \left( {H^{(i)}}^{-1}(\boldsymbol{\alpha}) - \boldsymbol{V}\boldsymbol{k} \right) \mathrm{d}\boldsymbol{\alpha}$$

- Relate $g^{(i)}(\boldsymbol{x})$ to $f(\boldsymbol{k})$ via LSV equations

$$g^{(i)}(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot \rho^{(i)}(\boldsymbol{x}, \boldsymbol{k})$$

- Change of variables $\boldsymbol{\alpha} = H^{(i)}(\boldsymbol{u})$:

$$\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k}) = \int h^{(i)} \left( \boldsymbol{x}, H^{(i)}(\boldsymbol{u}) \right) \cdot h_r \left( \boldsymbol{u} - \boldsymbol{V}\boldsymbol{k} \right) \left| \frac{\partial H^{(i)}}{\partial \boldsymbol{u}} \right| \mathrm{d}\boldsymbol{u}$$

  - expressed i.t.o warped PSF $h^{(i)}(\boldsymbol{x}, \boldsymbol{\alpha})$
  - integrate in $\boldsymbol{u}$-space (restoration)

# Multi-Frame Observation Model & Image Resampling

Multi-frame observation model:

$$g^{(i)}(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \int h^{(i)}(\boldsymbol{x},\boldsymbol{\alpha}) \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot h_r\left(H^{(i)^{-1}}(\boldsymbol{\alpha}) - \boldsymbol{V}\boldsymbol{k}\right) \mathrm{d}\boldsymbol{\alpha}$$

Image resampling:

$$g(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \int p(\boldsymbol{x}-\boldsymbol{\alpha}) \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{k}) \cdot r\left(H^{-1}(\boldsymbol{\alpha}) - \boldsymbol{k}\right) \mathrm{d}\boldsymbol{\alpha}$$

# Multi-Frame Observation Model & Image Resampling

| Multi-frame Observation model | | Image resampling | |
|---|---|---|---|
| Discrete scene estimate | $f(\boldsymbol{u})$ | Discrete texture | $f(\boldsymbol{u})$ |
| Interpolation kernel | $h_r(\boldsymbol{u})$ | Reconstruction filter | $r(\boldsymbol{u})$ |
| Scene/camera motion | $H^{(i)}(\boldsymbol{u})$ | Geometric transform | $H(\boldsymbol{u})$ |
| Observation SVPSF | $h^{(i)}(\boldsymbol{x}, \boldsymbol{\alpha})$ | Anti-alias pre-filter | $p(\boldsymbol{x})$ |
| Observed images | $g^{(i)}(\boldsymbol{x})$ | Warped output image | $g(\boldsymbol{x})$ |

Observation filter: $\rho^{(i)}(\boldsymbol{x}, \boldsymbol{k}) = \int h^{(i)}\left(\boldsymbol{x}, H^{(i)}(\boldsymbol{u})\right) \cdot h_r\left(\boldsymbol{u} - \boldsymbol{V}\boldsymbol{k}\right) \left|\frac{\partial H^{(i)}}{\partial \boldsymbol{u}}\right| \mathrm{d}\boldsymbol{u}$

Resampling filter : $\rho(\boldsymbol{x}, \boldsymbol{k}) = \int p\left(\boldsymbol{x} - H(\boldsymbol{u})\right) \cdot r\left(\boldsymbol{u} - \boldsymbol{k}\right) \left|\frac{\partial H}{\partial \boldsymbol{u}}\right| \mathrm{d}\boldsymbol{u}$

# Linear Multi-Frame Observation Model

- Recall Linear Shift Varying observation equation

$$g^{(i)}(\boldsymbol{x})\Big|_{\boldsymbol{x}\in\mathbb{Z}^2} = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\boldsymbol{k}) \cdot \rho^{(i)}(\boldsymbol{x},\boldsymbol{k})$$

- Admits matrix-vector representation in finite case
- Single row of observation matrix: (observed images have $N_r$ rows $\times N_c$ cols)

$$\boldsymbol{g}_j^{(i)} = \sum_{k=1}^{Q_y N_r Q_x N_c} A_{jk}^{(i)} \boldsymbol{f}_k$$

- Matrix-vector representation (single observed image)

$$\boldsymbol{g}^{(i)} = \boldsymbol{A}^{(i)} \boldsymbol{f}$$

## Linear Multi-Frame Observation Model

- Matrix-vector representation ($P$ images)

$$\boldsymbol{g} \doteq \begin{bmatrix} \boldsymbol{g}^{(1)} \\ \boldsymbol{g}^{(2)} \\ \vdots \\ \boldsymbol{g}^{(P)} \end{bmatrix} \quad \text{and} \quad \boldsymbol{A} \doteq \begin{bmatrix} \boldsymbol{A}^{(1)} \\ \boldsymbol{A}^{(2)} \\ \vdots \\ \boldsymbol{A}^{(P)} \end{bmatrix}$$

- Compact form

$$\boldsymbol{g} = \boldsymbol{A}\boldsymbol{f}$$

# A Bayesian Framework for Restoration

- Classic linear inverse problem
- Usually underconstrained (too few observations)
- Ill-posed, so use regularized solution method with *a-priori* information
- Augment observation model to include noise

$$\boldsymbol{g} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{n}$$

- Noise model is zero-mean Gaussian

$$\mathcal{P}_{\boldsymbol{N}}\left(\boldsymbol{n}\right) = \frac{1}{(2\pi)^{\frac{PN_rN_c}{2}}\left|\boldsymbol{K}\right|}\exp\left\{-\frac{1}{2}\boldsymbol{n}^T\boldsymbol{K}^{-1}\boldsymbol{n}\right\}$$

$\boldsymbol{K}$ is the p.d. covariance matrix.

# A Bayesian Framework for Restoration

- Noise model is zero-mean Gaussian

$$\mathcal{P}_{\boldsymbol{N}}\left(\boldsymbol{n}\right) = \frac{1}{(2\pi)^{\frac{PN_rN_c}{2}}\left|\boldsymbol{K}\right|} \exp\left\{-\frac{1}{2}\boldsymbol{n}^T\boldsymbol{K}^{-1}\boldsymbol{n}\right\}$$

$\boldsymbol{K}$ is the p.d. covariance matrix.

- Likelihood term

$$\begin{aligned}\mathcal{P}\left(\boldsymbol{g}|\boldsymbol{f}\right) &= \mathcal{P}_{\boldsymbol{N}}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f}) \\ &= \frac{1}{(2\pi)^{\frac{PN_rN_c}{2}}\left|\boldsymbol{K}\right|} \exp\left\{-\frac{1}{2}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})^T\boldsymbol{K}^{-1}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})\right\}\end{aligned}$$

# A Bayesian Framework for Restoration

- Prior term (Markov Random Field)
- Density is Gibbsian (Hammersley-Clifford)

$$\mathcal{P}\left(\boldsymbol{f}\right) = \frac{1}{k_p} \exp\left\{ -\frac{1}{\beta} \sum_{c \in \mathcal{C}} \rho_T\left(\boldsymbol{\partial}_c \boldsymbol{f}\right) \right\}$$

- Huber penalty function $\rho_T(x)$ (edge preserving)
- Local interactions $\boldsymbol{\partial}_c$ approximate $2^{\text{nd}}$ spatial derivates in 4 orientations

# A Bayesian Framework for Restoration

- Combined objective function

$$\begin{aligned}\hat{\boldsymbol{f}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{f}} \left\{ -\frac{1}{2}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})^T \boldsymbol{K}^{-1}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f}) - \frac{1}{\beta}\sum_{c\in\mathcal{C}} \rho_T\left(\boldsymbol{\partial}_c \boldsymbol{f}\right)\right\} \\ &= \arg\min_{\boldsymbol{f}} \left\{ \frac{1}{2}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f})^T \boldsymbol{K}^{-1}(\boldsymbol{g}-\boldsymbol{A}\boldsymbol{f}) + \gamma\sum_{c\in\mathcal{C}} \rho_T\left(\boldsymbol{\partial}_c \boldsymbol{f}\right)\right\}\end{aligned}$$

- Use your favorite optimization technique to find $\hat{\boldsymbol{f}}_{\mathrm{MAP}}$
- Unique solution under very mild conditions

# Simulation Details

| | |
|---|---|
| Projection model | Ideal pinhole |
| Image array dimensions | $128 \times 128$ |
| Pixel dimensions | $9\mu$m $\times$ $9\mu$m |
| Camera focal length | $10$ mm |
| Camera f/number | $2.8$ |
| Illumination wavelength | $550$ nm |
| Diffraction limit cutoff | $649.351$ cycles/mm |
| Sampling rate | $111.111$ samples/mm |
| Folding frequency | $55.5556$ cycles/mm |

Table 1: Camera intrinsic characteristics.

# Simulation Details

| Camera center | | | Camera gaze point | | |
|---|---|---|---|---|---|
| $x$ | $y$ | $z$ | $x$ | $y$ | $z$ |
| -3.0902 | -5.0000 | 9.5106 | 0.0100 | 0.0050 | 0.0000 |
| -1.0453 | -5.0000 | 9.9452 | 0.0033 | 0.0017 | 0.0000 |
| 1.0453 | -5.0000 | 9.9452 | -0.0033 | -0.0017 | 0.0000 |
| 3.0902 | -5.0000 | 9.5106 | -0.0100 | -0.0050 | 0.0000 |
| 5.0000 | -5.0000 | 8.6603 | -0.0167 | -0.0083 | 0.0000 |
| 6.6913 | -5.0000 | 7.4315 | -0.0233 | -0.0117 | 0.0000 |
| 8.0902 | -5.0000 | 5.8779 | -0.0300 | -0.0150 | 0.0000 |

Table 2: Camera extrinsic parameters.