Image Interpolation Using Kernel Regression

Presenter: Roy Wang

Outline

- Background
- Top-level view
- Theory – Classical Kernel Regression
  - Assumptions
  - Model: Linear Regression and Taylor expansion
  - Optimization: weighted least-squares formulation
- Example: Interpolation
- Theory – Steering Kernel Regression
  - Image orientation estimation using principal component analysis
  - Steering kernel image interpolation framework
- More Examples
  - Inpainting (missing data)
  - Denoise
- Discussion
  - Emphasis on quadratic objective function
  - Relevance for digital video processing research
- Summary

Background (1) – Kernel Regression

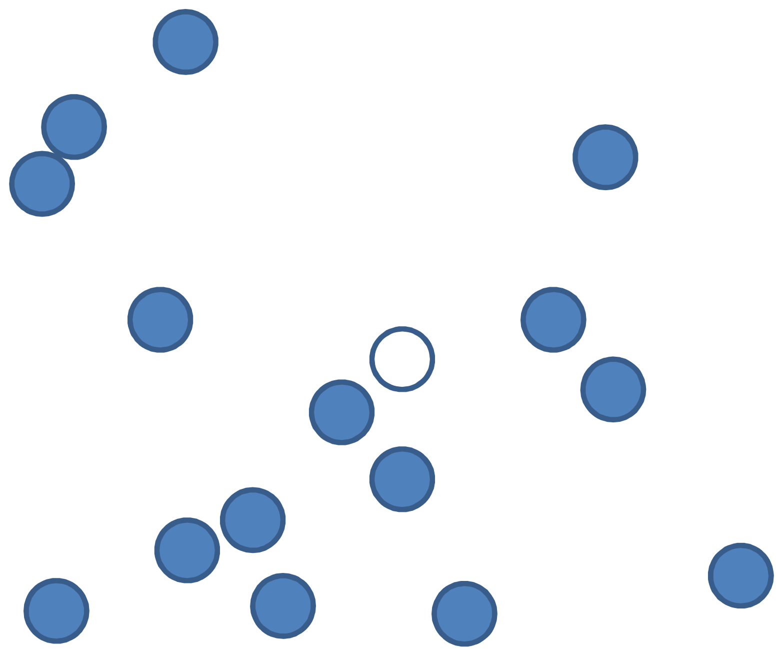
- Kernel Regression (KR)
  - Popularized for image processing by Hiro Takeda (supervised by Peyman Milanfar) from University of California Santa Cruz
  - Dated technique (2007)
- Interesting because…
  - Simple
  - Proposed a complete image/video regression model and framework
  - Model was extended in 2010 by Takeda for video data
    - Applied to video up-conversion research.

Background (2) – Similar Concepts

- Takeda's flavour of KR is similar to:
  - Moving least-squares
    - Reconstruct continuous surface from unorganized points (e.g. point cloud)
  - Normalized convolution
    - Generate regularly spaced points from irregularly sampled data, and then perform convolution
  - Bilateral filter
    - Noise-reduction filter that preserve edges, uses Gaussian and Euclidean distance for weights
  - Edge-directed interpolation
    - Anisotropic filtering that applies smooth data along "edges"

Top-Level View

- Takeda's image interpolation KR will be covered in this presentation
- Each sampled measurement y_i is represented by $z(x)$, $\nabla z(x)$, $H\{z(x)\}$, … (higher order terms), as well as the relative distance from the position of interest x to each of the y_i

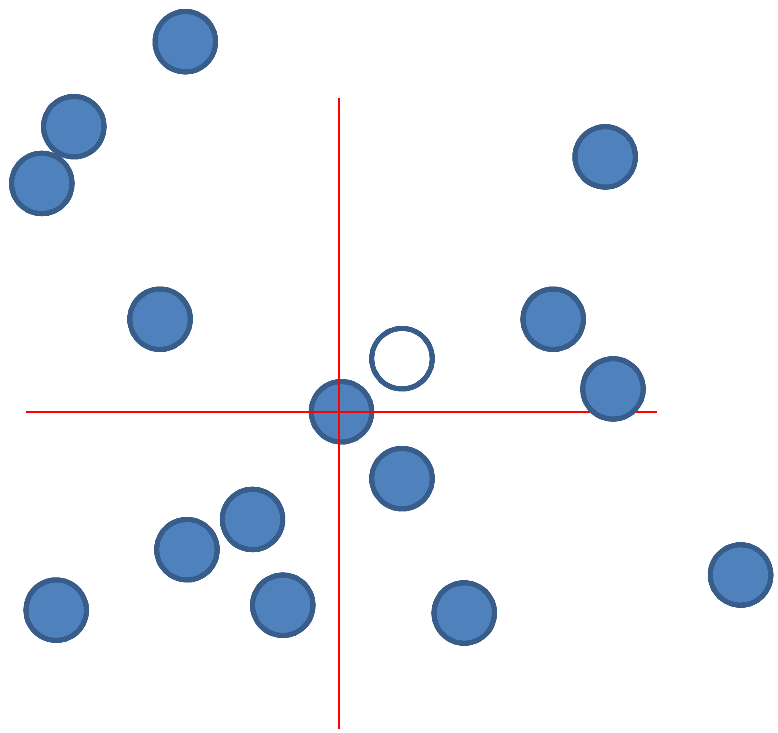


The position of the sampled measurements are solid

The hollow point is the position to perform interpolation

Top-Level View

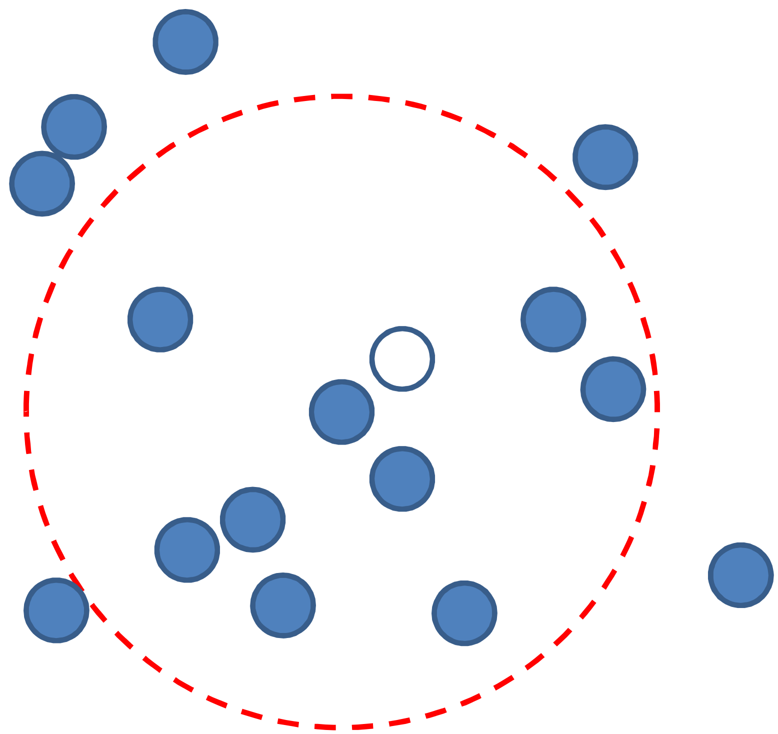
- Takeda's image interpolation KR will be covered in this presentation
- Each sampled measurement y_i is represented by $z(x)$, $\nabla z(x)$, $H\{z(x)\}$, … (higher order terms), as well as the relative distance from the position of interest x to each of the y_i



Define an analysis window

Top-Level View

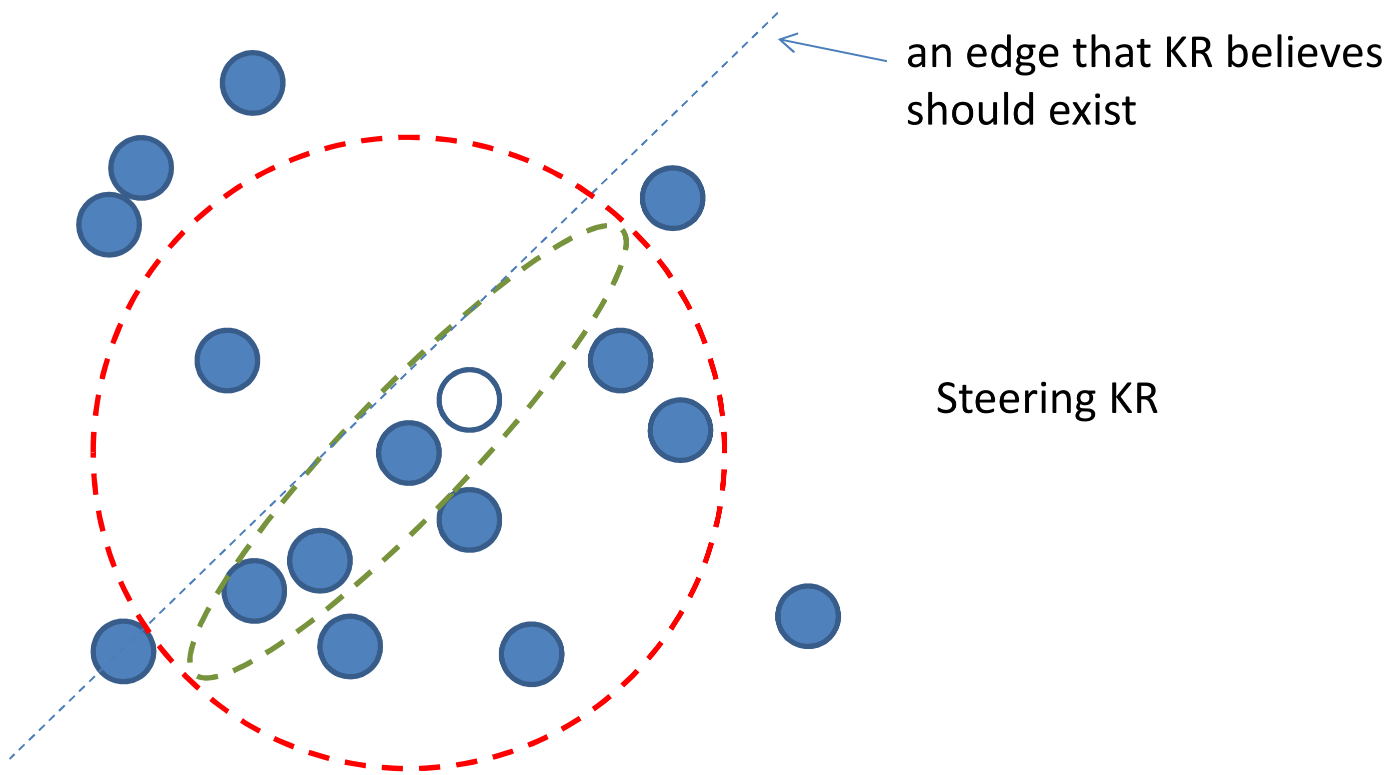
- Takeda's image interpolation KR will be covered in this presentation
- Each sampled measurement y_i is represented by $z(x)$, $\nabla z(x)$, $H\{z(x)\}$, … (higher order terms), as well as the relative distance from the position of interest x to each of the y_i



Classical KR

Top-Level View

- Takeda's image interpolation KR will be covered in this presentation
- Each sampled measurement y_i is represented by $z(x)$, $\nabla z(x)$, $H\{z(x)\}$, … (higher order terms), as well as the relative distance from the position of interest x to each of the y_i



KR Theory - Assumptions

- There exists a continuous signal upon which the input data were sampled from
  - Otherwise continuous mathematics cannot be justified to operate on a set of data points
- This continuous signal is assumed to
  - Accurately describe the data
  - Real and infinitely differentiable
    - Otherwise Taylor’s theorem does not hold
- The analysis window chosen for a coordinate x is a valid Taylor local neighbourhood to the continuous underlying signal

KR Theory – Regression Model

- Regression with additive error
- $y_i = z(x_i) + \varepsilon_i \quad , i = 1, \dots, P$
  - Measured data is $y_i \in \mathbb{R}$
  - Coordinates of y_i is $x_i \in \mathbb{R}^2$
  - P is the # of measured data in the neighbourhood
  - Model discrepancy error is $\varepsilon_i \in \mathbb{R}$
  - Regression function is $z(\cdot)$
  - $z(\cdot)$ can be non-linear, but it should be representable as a linear combination of some chosen basis

KR Theory – Taylor Expansion

- 1D Taylor's theorem for real analytic functions:
  $f(t) = f(a) + f'(a)(t-a) + \frac{f''(a)(t-a)^2}{2} + \cdots$
  - a is some point in the neighbourhood of t
- Apply the theorem to the 2D regression:

$$\begin{aligned} z(x_i) = z(x) \\ &+ \{\nabla z(x)\}^T (x_i - x) \\ &+ (x_i - x)^T H\{z(x)\}(x_i - x) \\ &+ \cdots \end{aligned}$$

- The goal is to estimate $z(x)$, $\nabla z(x)$,$H\{z(x)\}$, …
  - In the 1D case, $f(a)$, $f'(a)$, $f''(a)$, … is to be estimated.
  - For signal interpolation, ε is assumed to be a white and zero-mean noise, and $z(x)$ is assigned to be the data value y at coordinate x
  - A second order kernel regression would use a Taylor approx. of order 2
    - i.e. estimate only $z(x)$, $\nabla z(x)$, and $H\{z(x)\}$

KR Theory – Compact Representation

- After some matrix manipulation, the Taylor expansion of $z(x)$ could be written as

$$z(x_i) = z(x) + \{\nabla z(x)\}^T (x_i - x) + (x_i - x)^T H\{z(x)\}(x_i - x) + \cdots$$

$$= \beta_0 + \beta_1^T (x_i - x) + \beta_2^T \, vech\{(x_i - x)(x_i - x)^T\} + \cdots$$

where $vech\left(\begin{bmatrix} a & b \\ b & d \end{bmatrix}\right) = \begin{bmatrix} a & b & d \end{bmatrix}^T$

- Parameters of interest for estimation are
  - β_0, the predicted signal value at coordinate x
  - $\beta_1 = \nabla z(x) = \begin{bmatrix} \frac{\partial z(x)}{\partial x_1} & \frac{\partial z(x)}{\partial x_2} \end{bmatrix}^T$, the signal's gradient at coordinate x
  - $\beta_2 = \frac{1}{2}\begin{bmatrix} \frac{\partial^2 z(x)}{\partial x_1^2} & 2\frac{\partial^2 z(x)}{\partial x_1 \partial x_2} & \frac{\partial^2 z(x)}{\partial x_2^2} \end{bmatrix}^T$, components of the Hessian

KR Theory – Optimization Formulation

- Formulate the N -order estimation of $\{\beta_n\}_{n=0}^{N}$ as an optimization problem.
- Assume sample measurements y that are farther away from the desired coordinate x will be less significant in estimation of $z(x)$.
  - Represent this by using a weights on each of the y_i's
- Takeda chose to use a weighted least-squares approach .

$$min._{\{\beta_n\}} \sum_{i=1}^{P} [y_i - \beta_0 + \beta_1^T (x_i - x) + \beta_2^T\, vech\{(x_i - x)(x_i - x)^T\} + \cdots\ ]^2\, K_S(x_i - x)$$

- The kernel function $K_S(t)$ was chosen to be a Gaussian centered at t, and S modifies the support of K by a linear transformation S.
  - i.e. K can be scaled and rotated by the matrix S
- This formulation has an analytical solution for $\{\beta_n\}_{n=0}^{N}$
- Takeda referred to the case of $S = I$ as *classical KR*.

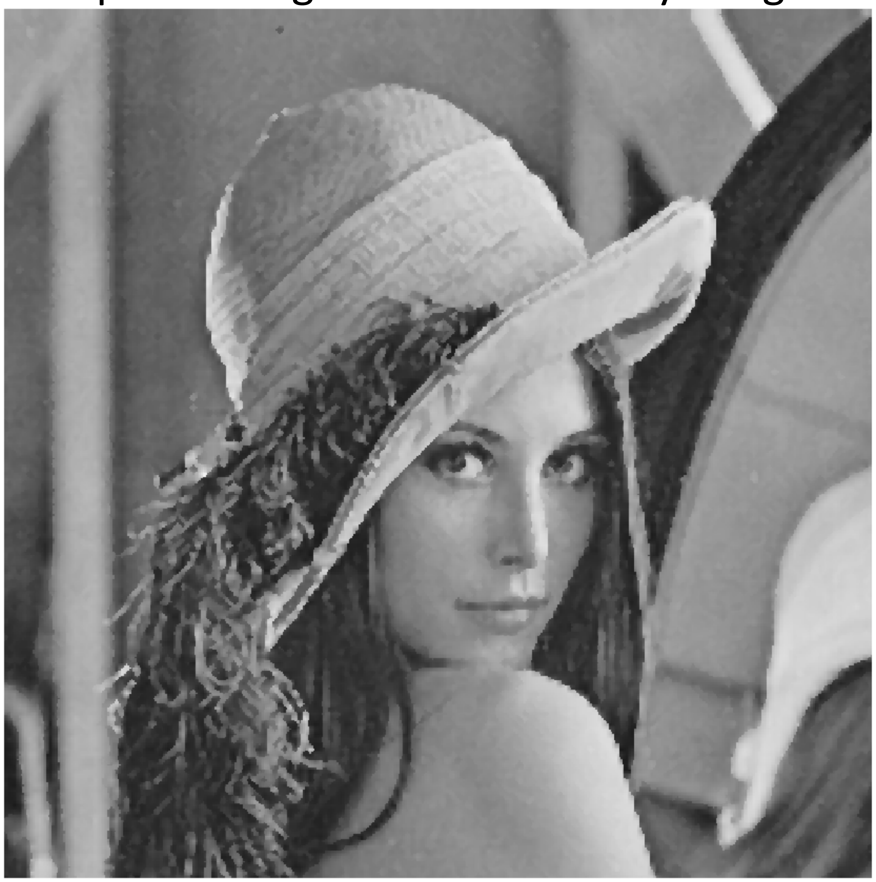
Examples (1) – Interpolation (1)

Examples (1) – Interpolation (2)

Examples (1) – Interpolation (3)

Examples (1) – Interpolation (4)

Improved Edge-directed: Jeremy Ranger



Examples (1) – Interpolation (5)

Statistical: Jeremy Ranger

Examples (1) – Interpolation (6)

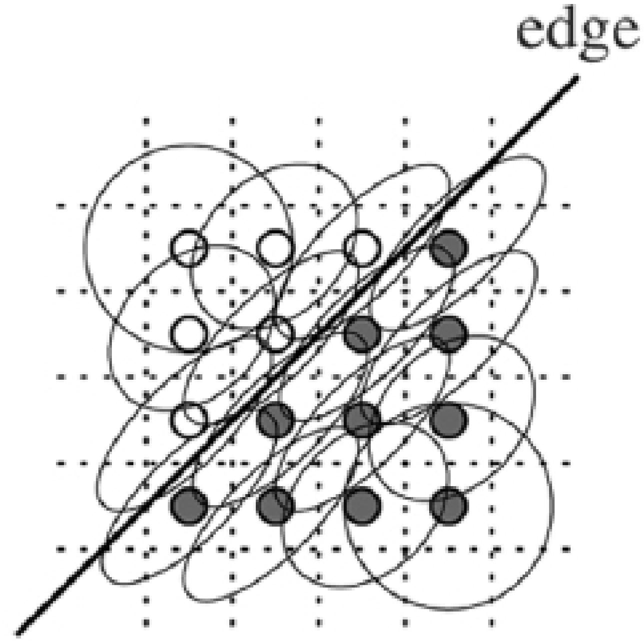
Total variation: Hussein Aly

Examples (1) – Interpolation (7)

Total variation: Hussein Aly

SKR Theory – Img. Orientation Est. (1)

- Assumes the local signal orientation of the data should be perpendicular to the gradient of each of the sampled measurement y's
  - Think "orientation" as "direction parallel to edges"
- Maximize the average angle between the orientation vector and the gradient



Takeda et. al (2007)

SKR Theory – Img. Orientation Est. (2)

- Max. average angle of all sample measurements within the analysis window
- Least-squares formulation of task yields a convex optimization problem:
- $\sum_{x_j \in W_i} (a^T g_i)^2 = a^T \sum_{x_j \in W_i} \left(g_i g_i^T \right) a = a^T D_i a$
  - Where g_i is $\nabla z\left(x_j\right)$, the analysis window W_i is centered at the position of the current sample under consideration y_i
  - $a \in \mathbb{R}^2$ is the orientation vector

$$\begin{array}{ll} max._a & a^T D_i a \\ subject\ to & a^T a = 1 \end{array}$$

- Normalized norm condition added for simple solution

SKR Theory – Lagrange Multipliers

- Lagrange multipliers
- $L(a, \lambda) = a^T D_i a - \lambda(a^T a - 1)$
- Set $\frac{\partial L}{\partial a} = 0$ gives $D_i a = \lambda a$
  - a must be an eigenvector of D_i that corresponds to λ, an eigenvalue of D_i
- Then $a^T D_i a = a^T \lambda a = \lambda$

  The max. occurs when λ is the largest eigenvalue
- Singular-value decomposition (SVD) can find λ and a

SKR Theory – Compute Eigenvectors

$$D_i = \sum_{x_j \in W_i} \nabla z(x_i)[\nabla z(x_i)]^T$$

$$= \begin{bmatrix} \sum_{x_j \in W_i} \left(\frac{\partial z(x_j)}{\partial e_1}\right)^2 & \sum_{x_j \in W_i} \frac{\partial z(x_j)}{\partial e_1}\frac{\partial z(x_j)}{\partial e_2} \\ \sum_{x_j \in W_i} \frac{\partial z(x_j)}{\partial e_2}\frac{\partial z(x_j)}{\partial e_1} & \sum_{x_j \in W_i} \left(\frac{\partial z(x_j)}{\partial e_2}\right)^2 \end{bmatrix}$$

$$= \begin{bmatrix} \cdots & \frac{\partial z\left(x_{j \in W_i}\right)}{\partial e_1} & \cdots \\ \cdots & \frac{\partial z\left(x_{j \in W_i}\right)}{\partial e_2} & \cdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots \\ \frac{\partial z\left(x_{j \in W_i}\right)}{\partial e_1} & \frac{\partial z\left(x_{j \in W_i}\right)}{\partial e_2} \\ \vdots & \vdots \end{bmatrix} \stackrel{\text{def}}{=} M^T M$$

- e_1 and e_2: horizontal and vertical spatial directions, respectively
- Fact: The right singular vectors of M (call it V)from a SVD are the eigenvectors of $M^T M$

SKR Theory – "Steering" Support

- To modify the original support of the kernel function K such that it "steers" along the orientation of the local data
  - Takeda's work had the original support of K being circular in shape
  - Would need an elliptical shape along the edges after the steering operation
    - Rotate and scale the function support of K
- Use tools from algebra; let:
  - $T: U \rightarrow U$ be a linear transformation from a vector space to itself,
  - $B = \{b_1, \ldots, b_n\}$ and $E = \{e_1, \ldots, e_n\}$ are different basis sets for U
- Then the matrices of T w.r.t. A and w.r.t. E are related by:

$$[T(E)]_E = [B]_E [T(B)]_B [B]_E^{-1}$$

  - (Note): Two square matrices F and N are said to be *similar* if there is an invertible P, such that $N = PFP^{-1}$

SKR Theory – “Steering” Support

$$[T(E)]_E = [B]_E[T(B)]_B[B]_E^{-1}$$

- Let T represent the entire function support steering operation (rotation and scaling of original support), and let B be an orthonormal basis set that have one of its basis vectors parallel to the estimated local signal orientation vector a
  - In other words, let B be the right singular vectors, V from the previous SVD decomposition of D_i
- Then T could then be an intuitive diagonal scaling matrix, that is meant to operate on the basis V:

$$[T(E)]_E \stackrel{\text{def}}{=} C_i = \gamma_i V_i \begin{bmatrix} \rho_i & 0 \\ 0 & \frac{1}{\rho_i} \end{bmatrix} V_i^T$$

$\rho_i = \frac{\lambda_1 + r_1}{\lambda_2 + r_1}$, $\gamma_i = \left(\frac{\lambda_1 \lambda_2 + r_2}{P}\right)^\tau$, r_1, r_2 and τ are regularization parameters

- Takeda decided the scaling ρ_i of the oriented support should be a function of the λ‘s of the SVD of D_i
  - Corresponds to the energy or confidence of the estimated orientation

SKR Theory – Steering Kernel

- Instead of using $S_i = I,\ \forall i$ as the linear transform on the kernel function, use $S_i = hC_i^{-\frac{1}{2}}$
  - to describe the ellipse as a rotated and scaled version of a circular support
- Takeda used a 2D Gaussian as the kernel function K. Written explicitly for this choice of S_i:

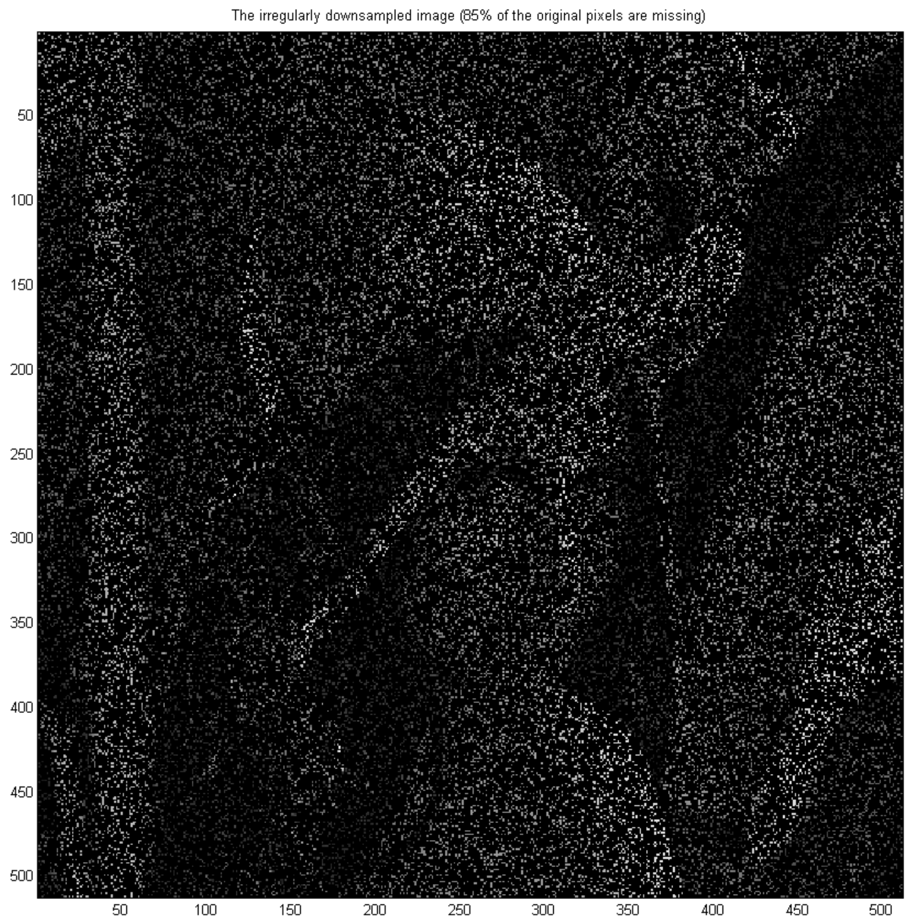
$$K_{S_i}(x_i - x) = \frac{\sqrt{det(C_i)}}{2\pi h^2} exp\left\{-\frac{(x_i - x)^T C_i (x_i - x)}{2h^2}\right\}$$

- This looks very similar to a bivariate Gaussian distribution with it's "footprint" controlled by a covariance matrix

SKR Theory – Algorithm Procedure

- Center the analysis window to the nearest sample measurement y_a to the desired interpolation coordinate x
  - The coordinate origin of the window would be located at the position coordinate of y_a. Call this position x_a.
- Obtain an estimate of the gradient of the y's within the window
  - Run classical KR but pass x_a as the interpolation coordinate.
  - This essentially estimates the betas of all y's within the analysis window.
  - Store the β_1's (gradient term) of these y's .
- Compute and store the steering kernel support matrix C_i for all y's in the entire image
- For each interpolation positions x, perform steering KR by using the classical KR but with the weight (kernel) function K computed from the stored C_i that corresponds to the y's within the local analysis window.

Example (2) – Inpainting (1)



Example (2) – Inpainting (2)

Example (2) – Inpainting (2)

- Note: I wonder how this relates to compressed sensing?

Example (3) - Denoise

The original image

The denoised image by iterative steering kernel regression, 3 iterations

Discussion (1)

- This framework is a typical regression problem.
  - The regression function could used other forms of image representation tools (i.e. other basis) for specific types of images
  - However, the use of Taylor approximation allowed both the signal value and its nth-order derivatives to be jointly estimated
    - This is desirable for applications where motion estimation (ME) algorithms are often inaccurate
    - Errors in the ME algorithm would have less chance of creating visually unbearable up-conversion artefacts

Discussion (2)

- Euclidean distances were used in the optimization formulation
  - Quadratic objective functions heavily penalize outliers to the assumed model; thus regions of the image that are not well-represented by the Taylor approx. have trouble
- Could introduce a penalty term (the prior term in Markov image frameworks) in the optimization formulation at the expense of foregoing an analytical solution (the weighted least-squares)
  - Since Hussein Aly's total variation image magnification framework from 2004 was able to recover more details in certain situations

Summary

- The KR technique for digital image processing was introduced this tutorial seminar
  - This is a very clean and intuitive regression framework
- Even though the modelling of an image as a continuous analytical function is unrealistic, it allowed the use of Taylor approximation
  - The Taylor approximation does not require regularly sampled measurements
- One solution to the local signal orientation estimation problem of the was also introduced
  - Takeda used a least-squares formulation, with hints of PCA

References

- H. Takeda, P. van Beek, and P. Milanfar, "Spatiotemporal video upscaling using Motion-Assisted Steering Kernel (MASK) regression," in High-quality visual experience: creation, processing and interactivity of high-resolution and high-dimensional video signals (M. Mrak, M. Grgic, and M. Kunt, eds.), pp. 245–274, Springer, 2010
- H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," IEEE Transactions on Image Processing, vol. 16, no. 2, pp. 349–366, 2007.
- Jeremy Ranger and Hussein Aly's sample MATLAB code for image interpolation
- Hiro Takeda, "Kernel Regression-Based Image Processing Toolbox for MATLAB", http://users.soe.ucsc.edu/~htakeda/KernelToolBox.htm