

Kernel Regression Based Image Processing Toolbox for MATLAB

Version 1.2 β

Hiroyuki Takeda

Multi-Dimensional Signal Processing Laboratory
University of California, Santa Cruz

Directory Structure

- Kernel Regression
 - This directory contains the main functions of kernel regression.
- Support Functions
 - This directory contains the sub functions for the main functions.
- Examples
 - Some simulation scripts are available in this directory.
- Test Images
 - There are some test images in this directory.

Directory Structure

- Kernel Regression
 - This directory contains the main functions of kernel regression.
- Support Functions
 - This directory contains the sub functions for the main functions.
- Examples
 - Some simulation scripts are available in this directory.
- Test Images
 - There are some test images in this directory.

Kernel Regression Functions

- 9 functions of kernel regression and an orientation estimation function are available.
 - Second order classic kernel regression for regular and irregular data.
 - “ckr2_regular”, “ckr2all_regular”, “ckr2L1_regular” and “ckr2_irregular”
 - Zeroth order steering kernel regression for regular and irregular data.
 - “skr0_regular” and “skr0_irregular”
 - Second order steering kernel regression for regular and irregular data.
 - “skr2_regular”, “skr2L1_regular” and “skr2_irregular”
 - Orientation estimation function
 - “steering”

ckr2_regular

- Description
 - Second order classic kernel regression for regularly sampled data with Gaussian kernel function
- Usage
 - $[z, zx1, zx2] = \text{ckr2_regular}(y, h, r, \text{ksize})$
- Returns
 - z : the estimated image
 - zx1, zx2 : the estimated gradient images along x1 and x2 directions
- Parameters
 - y : the input image
 - h : the global smoothing parameter
 - r : the upscaling factor
 - ksize : the support size of the kernel function

ckr2all_regular

- Description
 - Second order classic kernel regression for regularly sampled data with Gaussian kernel function. This function returns the estimated image and all the first and second gradients.
- Usage
 - `z = ckr2all_regular(y, h, r, ksize)`
- Returns
 - `z` : the estimated image and all the first and second gradients.
- Parameters
 - `y` : the input image
 - `h` : the global smoothing parameter
 - `r` : the upscaling factor
 - `ksize` : the support size of the kernel function

ckr2L1_regular

- Description
 - Second order classic kernel regression with L1-norm for regularly sampled data with Gaussian kernel function. This function returns the estimated image and all the first and second gradients.
- Usage
 - `z = ckr2L1_regular(y, z_init, h, r, ksize, IT, step)`
- Returns
 - `z` : the estimated image and all the first and second gradients.
- Parameters
 - `y` : the input image
 - `z_init` : the initial state for the steepest descent method
 - `h` : the global smoothing parameter
 - `r` : the upscaling factor
 - `ksize` : the support size of the kernel function
 - `IT` : the number of iterations for steepest descent method
 - `step` : the step size of the steepest descent update

ckr2_irregular

- Description
 - Second order classic kernel regression for irregularly sampled data with Gaussian kernel function
- Usage
 - $[z, zx1, zx2] = \text{ckr2_irregular}(y, I, h, \text{ksize})$
- Returns
 - z : the estimated image
 - zx1, zx2 : the estimated gradient images along x1 and x2 directions
- Parameters
 - y : the input image
 - I : the sampling position map (1 where we have samples)
 - h : the global smoothing parameter
 - ksize : the support size of the kernel function

skr0_regular

- Description
 - Zeroth order steering kernel function for regularly sampled data with Gaussian kernel function
- Usage
 - $z = \text{skr0_regular}(y, h, C, r, \text{ksize})$
- Returns
 - z : the estimated image
- Parameters
 - y : the input image
 - h : the global smoothing parameter
 - C : the inverse covariance matrices which contain local orientation information
 - r : the upscaling factor
 - ksize : the support size of the kernel function

skr2_regular

- Description
 - Second order steering kernel function for regularly sampled data with Gaussian kernel function
- Usage
 - $[z, zx1, zx2] = \text{skr2_regular}(y, h, C, r, \text{ksize})$
- Returns
 - z : the estimated image
 - zx1, zx2 : the estimated gradient images along x1 and x2 directions
- Parameters
 - y : the input image
 - h : the global smoothing parameter
 - C : the inverse covariance matrices which contain local orientation information
 - r : the upscaling factor
 - ksize : the support size of the kernel function

skr2L1_regular

- Description
 - Second order steering kernel regression with L1-norm for regularly sampled data with Gaussian kernel function. This function returns the estimated image and all the first and second gradients.
- Usage
 - `z = ckr2L1_regular(y, z_init, h, C, r, ksize, IT, step)`
- Returns
 - `z` : the estimated image and all the first and second gradients.
- Parameters
 - `y` : the input image
 - `z_init` : the initial state for the steepest descent method
 - `h` : the global smoothing parameter
 - `C` : the inverse covariance matrices which contain local orientation information
 - `r` : the upscaling factor
 - `ksize` : the support size of the kernel function
 - `IT` : the number of iterations for steepest descent method
 - `step` : the step size of the steepest descent update

skr0_irregular

- Description
 - Zeroth order steering kernel regression function for irregularly sampled data with Gaussian kernel function
- Usage
 - $z = \text{skr0_irregular}(y, I, h, C, \text{ksize})$
- Returns
 - z : the estimated image
- Parameters
 - y : the input image
 - I : the sampling position map (1 where we have samples)
 - h : the global smoothing parameter
 - C : the inverse covariance matrices which contain local orientation information
 - ksize : the support size of the kernel function

skr2_irregular

- Description

- Second order steering kernel regression function for irregularly sampled data with Gaussian kernel function

- Usage

- $[z, zx1, zx2] = \text{skr2_irregular}(y, I, h, C, \text{ksize})$

- Returns

- z : the estimated image
- zx1, zx2 : the estimated gradient images along x1 and x2 directions

- Parameters

- y : the input image
- I : the sampling position map (1 where we have samples)
- h : the global smoothing parameter
- C : the inverse covariance matrices which contain local orientation information
- ksize : the support size of the kernel function

steering

- Description
 - Orientation estimation function using singular value decomposition for steering kernel regression
- Usage
 - $C = \text{steering}(zx1, zx2, I, wsize, lambda, alpha)$
- Returns
 - C : the inverse covariance matrices which contain local orientation information
- Parameters
 - $zx1, zx2$: the gradient images along $x1$ and $x2$ directions
 - I : the sampling position map (1 where we have samples)
 - $wsize$: the size of the local analysis window
 - $lambda$: the regularization for the elongation parameter
 - $alpha$: the structure sensitive parameter

Directory Structure

- Kernel Regression
 - This directory contains the main functions of kernel regression.
- Support Functions
 - This directory contains the sub functions for the main functions.
- Examples
 - Some simulation scripts are available in this directory.
- Test Images
 - There are some test images in this directory.

Examples

- 6 examples are available to show how to use the kernel regression functions.
 - “Lena_denoise.m”
 - Image denoising example using the algorithm of iterative steering kernel regression
 - “Lena_upscale.m”
 - Image upscaling example by steering kernel regression
 - “Lena_irregular.m”
 - Image reconstruction example from irregularly downsampled image by steering kernel regression
 - “Lena_saltpepper.m”
 - Salt & pepper noise reduction example.
 - “Pepper_deblock.m”
 - Compression artifact removal example using the algorithm of iterative steering kernel regression
 - “JFK_denoise.m”
 - Real denoising example for a color image (Film grain noise removal)

Summary

- The kernel regression framework is very easy to implement.
- Other simulations are also possible by using the function set such as color artifact reduction and simultaneous interpolation and denoising.

Relevant Publication

- Takeda, H., S. Farsiu, and P. Milanfar, “Kernel regression for image processing and reconstruction,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, February 2007.
- Takeda, H., S. Farsiu, and P. Milanfar, “Robust kernel regression for restoration and reconstruction of Images from sparse noisy data,” Proceedings of the International Conference on Image Processing (ICIP), Atlanta, GA, October 2006.