Guided Image Filtering with Arbitrary Window Function

*Norishige Fukushima† Kenjiro Sugimoto‡ Sei-ichiro Kamata‡

†Nagoya Institute of Technology, Japan ‡Waseda University, Japan *fukushima@nitech.ac.jp

Introduction

Background

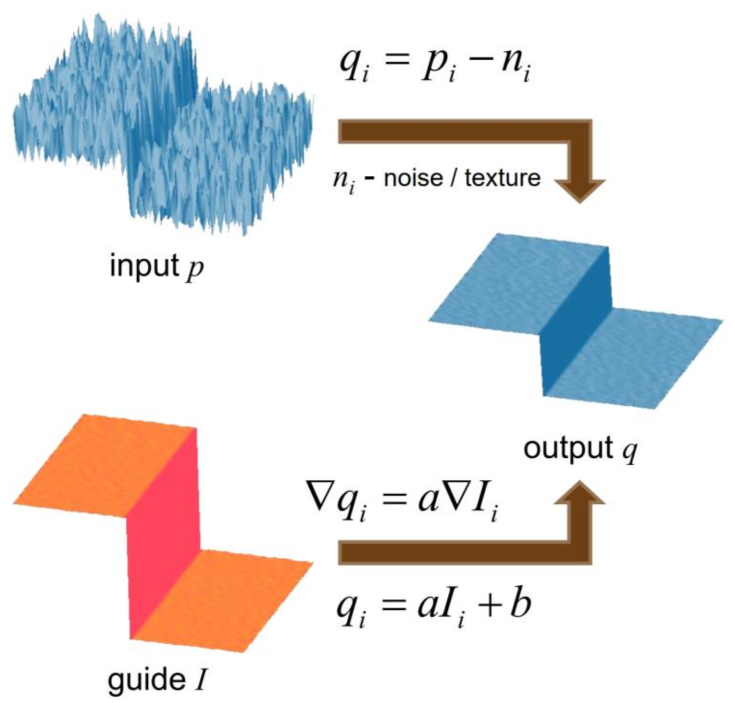
- Edge preserving filtering is essential tools for current image processing and computer vision.
  - denoising, detail enhancement, HDR, haze removing, stereo matching, optical flow, image coding.
- Guided image filtering (GIF) is a fast edge preserving filter.
  - constant time for filtering kernel radius
- limitation of guided image filtering is setting kernel shape

Contributions

1. Extending the definition of GIF for designing arbitrary kernel shape of filtering.
2. Keep constant time property of GIF.

Overview of guided image filter

1: $mean_I = f_{mean}(I, r)$
$corr_I = f_{mean}(I.*I, r)$
$var_I = corr_I - mean_I.*mean_I$
2: $mean_p = f_{mean}(p, r)$
$corr_{Ip} = f_{mean}(I.*p, r)$
$cov_{Ip} = corr_{Ip} - mean_I.*mean_p$
3: $a = cov_{Ip}./(var_I + \epsilon)$
$b = mean_I - a.*mean_I$
4: $mean_a = f_{mean}(a, r)$
$mean_b = f_{mean}(b, r)$
5: $q = mean_a.*I + mean_I$



- All computation consists of Hadamard product and simple box filtering (f_{mean}), which has constant time algorithm for filtering kernel radius.
- Computational time is not depend on filtering kernel.

Guided Image Filtering

Conventional

Assumption: an output patch q is a linear transform of a patch of a guidance image I.

$$q'_i = a_k I_i + b_k, \forall i \in \omega_k$$

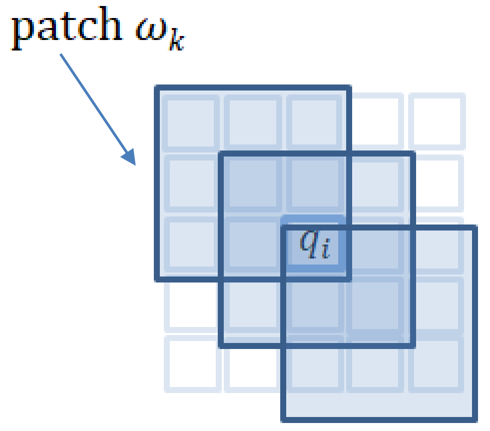
Coefficients a, b are introduced by linear regression between I and input image p.

$$\arg\min_{a_k, b_k} = \sum_{i \in \omega_k} ((a_k I_i + b_k - p_i)^2 + \epsilon a_k^2)$$

Solving this:

$$a_k = \frac{\mathrm{cov}_k(I, p)}{\mathrm{var}_k(I) + \epsilon}$$

$$b_k = \bar{p}_k - a_k \bar{I}_k$$



Averaging each patch for output image.

$$q_i = \frac{1}{|\omega|} \sum_{k | i \in \omega_k} (a_k I_i + b_k)$$
$$= \bar{a}_i I_i + \bar{b}_i,$$

The per patch mean deforms averages of coefficients.

(a) box (b) Gauss (c) d-exp

Visualized kernel weight.

Arbitrary windowed

Assumption: we use weighted linear regression of the whole image, instead of each local patch.

・e.g., emphasize focusing a pixel.

・patch operation is equal to a square window for an image.

$$\arg\min_{a_k, b_k} = \sum_{i \in \Omega} w_{i,k}((a_k I_i + b_k - p_i)^2 + \epsilon a_k^2)$$

Solving this:

$$a_k = \frac{\hat{\mathrm{cov}}_k(I, p)}{\hat{\mathrm{var}}_k(I) + \epsilon} \quad \hat{x}_k = \frac{\sum_{i \in \Omega} w_{k,i} x_i}{\sum_{i \in \Omega} w_{k,i}}$$

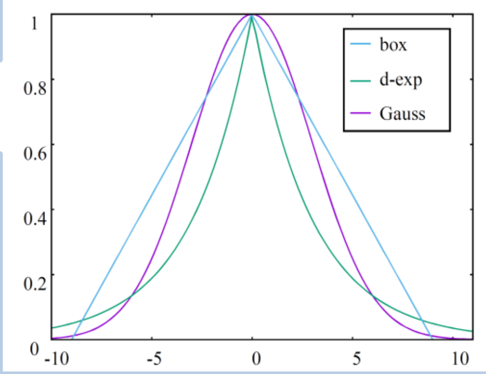
$$b_k = \hat{p}_k - a_k \hat{I}_k$$

where hat means weighted average.

Finally, we average whole converted images. The equation also deforms weighted averages of coefficients.

$$q_i = \frac{\sum_{k \in \Omega} w_{i,k}(a_k I_i + b_k)}{\sum_{k \in \Omega} w_{i,k}}$$
$$= \hat{a}_i I_i + \hat{b}_i$$

- ☐ Weighted linear regression can be represented by arbitrary windowed image filtering.
- ☐ If the weighted mean is constant time filter, the extended GIF is also constant time. For example,
  - IIR filter (Gaussian, dual exponential (d-exp))
  - constant time FIR filter (box, Gaussian)
  - constant time bilateral filter
  - guided filter itself (recursive applying)



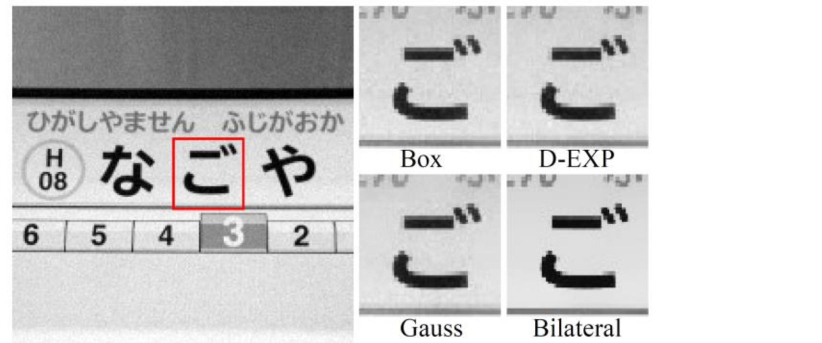
- box $w_{i,k} = 1$
- Gauss $w_{i,k} = \exp(-\frac{\|i-k\|^2}{2\sigma^2})$
- d-exp $w_{i,k} = \exp(-\frac{|i-k|}{\sigma})$

Experimental Results

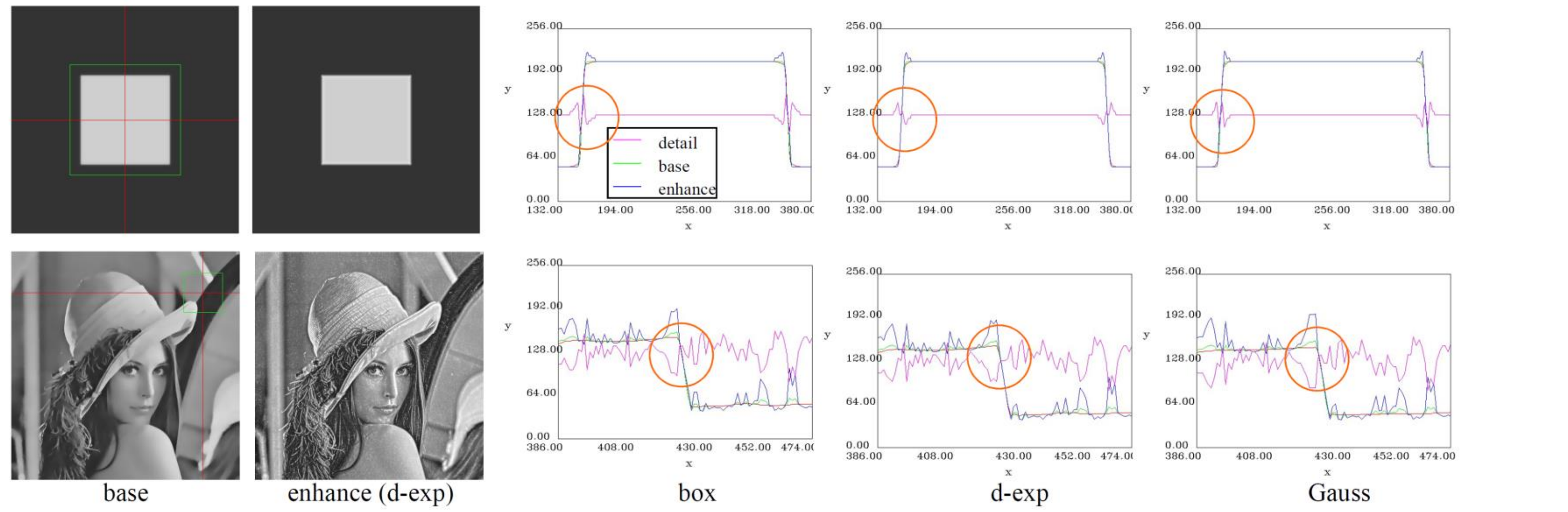
Variation of filters and applications

- ☐ detail enhancement
- ☐ denoising
- ☐ haze removing
- ☐ LTI filter: box, Gaussian, dual exponential smoothing
- ☐ LTV filter: bilateral filter

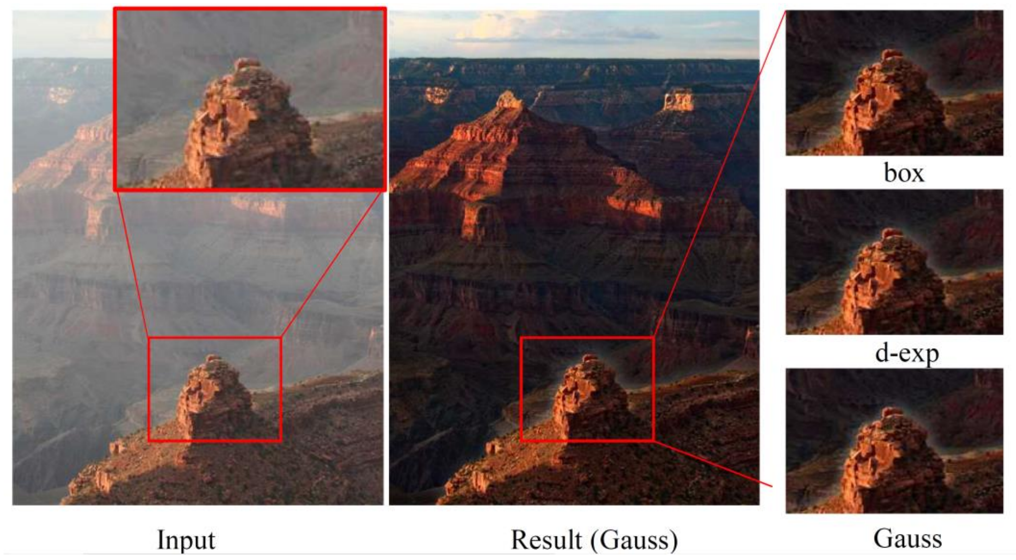
noise σ	box	d-exp	Gauss
5	37.68	37.31	**37.76**
10	33.31	33.26	**33.40**
15	31.50	31.14	**31.72**



Denoising : Gaussian filtering is the best. If the assumption of local linearity is not supported (the case of bimodal histogram), edge-preserving filter is better.



Detail enhancement: d-exp suppresses halo.



Haze remove: box filter is the best.

	box	d-exp*	Gauss
512x512	4.68	6.35	6.23
1024x1024	22.58	32.98	29.14
2048x2048	88.65	134.5	118.78

*faster than paper version.

Computational time [ms]