Near-Constant Time Bilateral Filter For High Dimensional Images

Christina Karam[1], Kenjiro Sugimoto[2], Keigo Hirakawa[1]

[1] University of Dayton, Electrical and Computer Engineering, {ckaram1,khirakawa1}@udayton.edu
[2] Waseda University, Graduate School of Information, Production and Systems, ksugimoto@aoni.waseda.jp

1. Goal Of This Work

Grayscale	Color	Hyperspectral
768x512x1	768x512x3	1392x1040x31

- Grayscale image has only intensity information at each pixel location ($C = 1$)
- Color image has red, green, and blue samples at each pixel ($C = 3$)
- Hyperspectral image records spectra at each pixel ($C \gg 3$)
- We previously found a way to accelerate high-dimensional bilateral filtering using stochastic filtering

We propose a faster stochastic filter that reduces the number of convolutions by $C + 1$ times.

2. Conventional Bilateral Filter (BF)

spatial filter kernel : $w_s(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|^2}{2\sigma_s^2}\right)$

range filter kernel : $w_r(\boldsymbol{g}(\boldsymbol{x}), \boldsymbol{g}(\boldsymbol{y})) = \exp\left(-\frac{\|\boldsymbol{g}(\boldsymbol{x})-\boldsymbol{g}(\boldsymbol{y})\|^2}{2\sigma_r^2}\right)$

$$\widehat{\boldsymbol{f}}(\boldsymbol{x}) := \frac{\sum_{\boldsymbol{y}\in\mathbb{Z}^2} w_s(\boldsymbol{x}-\boldsymbol{y}) w_r(\boldsymbol{g}(\boldsymbol{x})-\boldsymbol{g}(\boldsymbol{y}))\boldsymbol{f}(\boldsymbol{y})}{\sum_{\boldsymbol{y}\in\mathbb{Z}^2} w_s(\boldsymbol{x}-\boldsymbol{y}) w_r(\boldsymbol{g}(\boldsymbol{x})-\boldsymbol{g}(\boldsymbol{y}))}.$$

3. Problems with Existing "Fast" Bilateral Filters

- Several Fast Bilateral Filters have already been developed
- Replaced slow weight computation by K fast convolutions (Chaudhury, 2011)
- Replaced slow weight computation by three dimensional convolutions and Q number of quantization steps (Paris, 2006)
- They are independent of window size W, but expensive for C.
- Stochastic filter replaces the range kernel for faster computation:

$$\mathbb{E}\left[\exp\left(\pm j \boldsymbol{X}^T(\boldsymbol{g}(\boldsymbol{x})-\boldsymbol{g}(\boldsymbol{y}))\right)\right] = \exp\left(-\frac{(\boldsymbol{g}(\boldsymbol{x})-\boldsymbol{g}(\boldsymbol{y}))^2}{2\sigma_r^2}\right),$$

with $\boldsymbol{X} \in \mathbb{R}^C$, $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}/\sigma_r^2)$ denote a normal random vector, and $\boldsymbol{g} \in \mathbb{R}^C$ is a constant vector

4. Proposed: Stochastic Compressive Bilateral Filter (SCBF)

Let $\boldsymbol{\zeta} \sim \mathcal{N}(0, \sigma_r^{-2}\boldsymbol{I})$, $\zeta \in \mathbb{R}^C$ be a normal random vector, where $\boldsymbol{I} \in \mathbb{R}^{C\times C}$ is an identity matrix.

Idea: Rewrite range filter kernel as: $w_r(\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{y})) = \mathbb{E}\left[\cos\left(\boldsymbol{\zeta}^T(\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{y}))\right)\right]$.

$$w_r(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})) = \mathbb{E}\left[\cos(\boldsymbol{\zeta}^T(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})))\right] = \mathbb{E}\left[\begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x})) & \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}\begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{y}))\\ \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{y}))\end{bmatrix}\right].$$

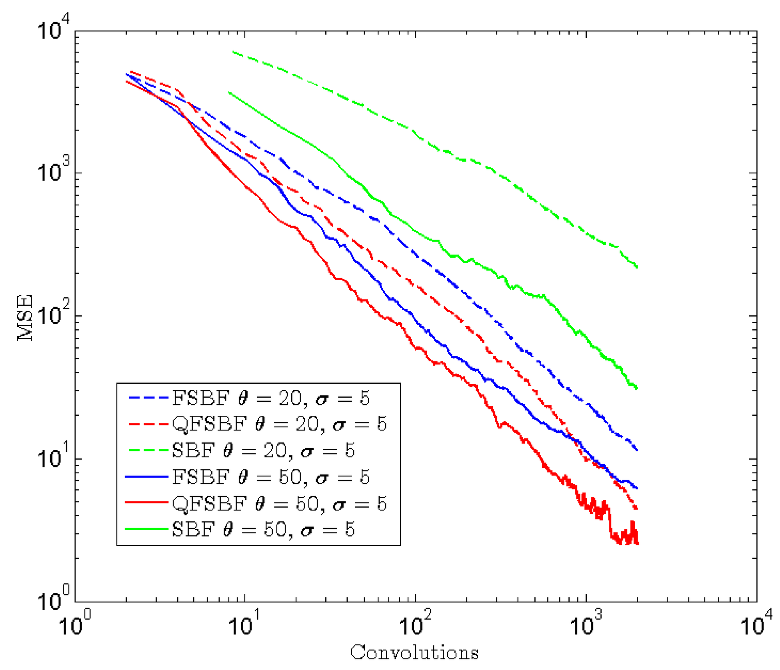
$$\nabla w_r(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})) = \mathbb{E}[-\boldsymbol{\zeta}\sin(\boldsymbol{\zeta}^T(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})))] = \mathbb{E}\left[\boldsymbol{\zeta}\begin{bmatrix}-\sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x})) & \cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}\begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{y}))\\ \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{y}))\end{bmatrix}\right]$$

$$\widehat{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}) + \sigma_r^2 \frac{\mathbb{E}\left[\boldsymbol{\zeta}\begin{bmatrix}-\sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\\ \cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}^T\left(w_s(\boldsymbol{x}) \star \begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\\ \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}\right)\right]}{\mathbb{E}\left[\begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\\ \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}^T\left(w_s(\boldsymbol{x}) \star \begin{bmatrix}\cos(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\\ \sin(\boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x}))\end{bmatrix}\right)\right]}.$$

- The convolution operator in the denominator and numerator are identical.
- Convolutions are one-dimensional.

Stochastic Compressive Bilateral Filter

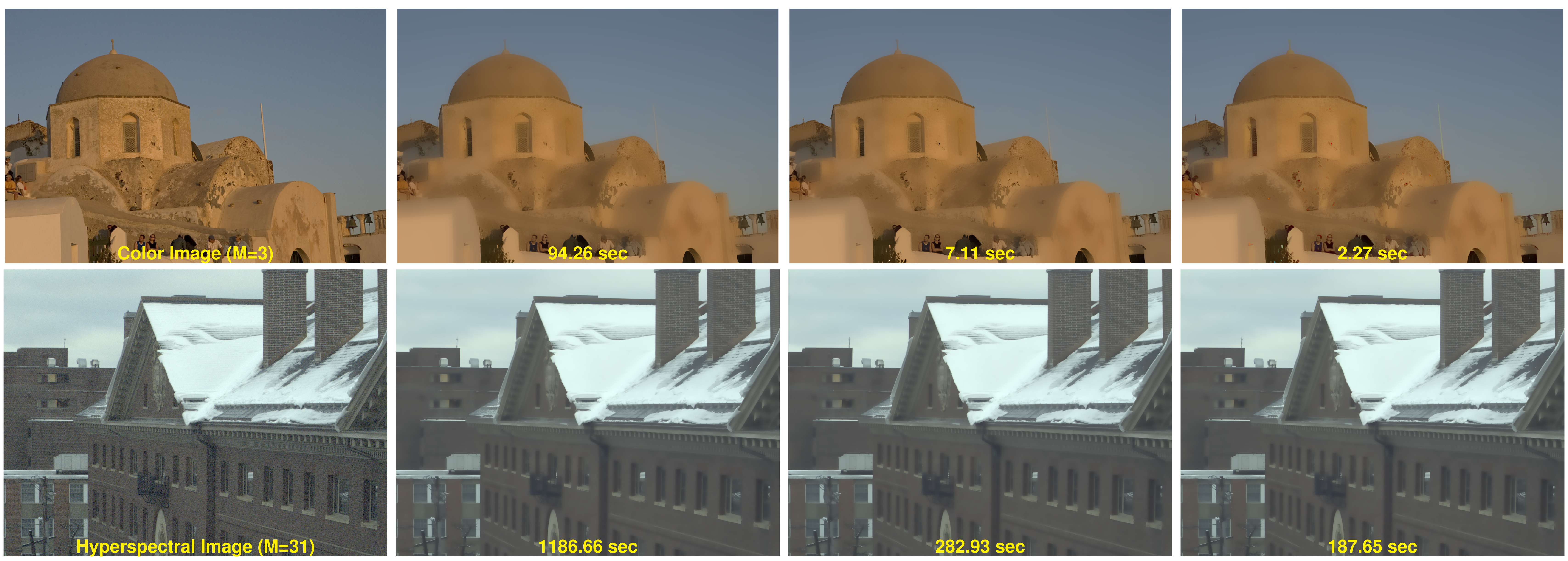
input: $\boldsymbol{f} : \mathbb{Z}^2 \to \mathbb{R}^C$
output: $\widehat{\boldsymbol{f}} : \mathbb{Z}^2 \to \mathbb{R}^C$
parameters: σ_r, σ_s
initialize numerator $\boldsymbol{n}(\boldsymbol{x}) \Leftarrow 0$ and denominator $d(\boldsymbol{x}) \Leftarrow 0$
for L times **do**
  generate $\boldsymbol{\zeta} \sim \mathcal{N}(\boldsymbol{0}, \sigma_r^{-2}\boldsymbol{I})$
  compute $z(\boldsymbol{x}) \Leftarrow \boldsymbol{\zeta}^T\boldsymbol{f}(\boldsymbol{x})$
  compute $c(\boldsymbol{x}) \Leftarrow \cos(z(\boldsymbol{x}))$ and $s(\boldsymbol{x}) = \sin(z(\boldsymbol{x}))$
  compute $\gamma(\boldsymbol{x}) \Leftarrow w_s(\boldsymbol{x}) \star c(\boldsymbol{x})$
  compute $\beta(\boldsymbol{x}) \Leftarrow w_s(\boldsymbol{x}) \star s(\boldsymbol{x})$
  update $\boldsymbol{n}(\boldsymbol{x}) \Leftarrow \boldsymbol{n}(\boldsymbol{x}) + \boldsymbol{\zeta}\left(\beta(\boldsymbol{x})c(\boldsymbol{x}) - \gamma(\boldsymbol{x})s(\boldsymbol{x})\right)$
  update $d(\boldsymbol{x}) \Leftarrow d(\boldsymbol{x}) + c(\boldsymbol{x})\gamma(\boldsymbol{x}) + s(\boldsymbol{x})\beta(\boldsymbol{x})$
end for
set $\widehat{\boldsymbol{f}}(\boldsymbol{x}) \Leftarrow \boldsymbol{f}(\boldsymbol{x}) + \sigma_r^2\boldsymbol{n}(\boldsymbol{x})/d(\boldsymbol{x})$



	per pixel					per image	
	multiply	add	divide	exp/sin/cos	memory	convolution	clusters
Original	$W^2(D+C+2)$	$W^2D+(D-1)+(C+1)(W^2-1)$	C	W^2	$1+D+C$	0	0
Paris	$(D+2)Q^D$	$(D+1)Q^D$	C	Q^D	CQ^D	2	0
Chaudhury Sugimoto	$(D+2C+1)K^D$	DK^D+2K^{D-1}	C	K^D	$1+D+C$	$(C+1)K^D$	0
Deng	$(3C+1)K^C$	$CK^C+2K^{C-1}+C$	C	$2K^C$	$1+2C$	K^C	0
Karam	$(D+4C+2)L$	$(D+C)L+(C+1)(L-1)$	C	$2L$	$1+D+C$	$(2C+2)L$	0
Sugimoto	$(C+1)K$	$KD+2(K-1)$	$C+1$	K	$1+D+C$	$(C+1)K$	K
Nair	CK	KD	C	K	$1+D+C$	$(C+1)K$	K
SCBF (proposed)	$(5C+2)L$	$(2C)L+(C+1)(L-1)+C$	C	$2L$	$1+2C$	$2L$	0

- L = number of iterations in SCBF
- Quasi-random sequence reduces L
- We generalize this for non-local means (paper under review)

5. Results



Noisy Edge Image e(i)	BF Result	SBF Result [1]	Proposed: SCBF Result

[1] C.Karam and K.Hirakawa, "Monte-Carlo acceleration of bilateral filter and non-local means," *IEEE Trans. Image Process.*, vol.27, no. 3, pp. 1462-1474, Mar. 2018

IEEE International Conference on Image Processing, October 2018, Athens