A little bit of (personal) history

Standard unified formulations (nonlinear filters) fail to capture some details, e.g. due to texture!

- mid 80’s – unified formulations – a breakthrough!
- methods that combine smoothing and edge detection (Geman & Geman’84, Mumford & Shah’89, Perona & Malik’90)
**Some seminal works**

- Total Variation Filter
- Fast Cartoon + Texture
- Relative Total Variation
- L0 Smoothing
- Context-guided Filtering
- RegCov Smoothing
- Rolling Guidance Filter
- Envelope Extraction

**Context-guided filtering**

- Contextual knowledge extracted from local image regions guides the regularization process.

**Structure-Texture Decomposition**

- Decomposing an image into structure and texture components

Input Image

Structure Component

Image Credit: P. Milanfar
Structure-Texture Decomposition

- Decomposing an image into structure and texture components

Texture Component

Region Covariances as Region Descriptors

Tuzel et al., ECCV 2006

\[
F(x,y) = \hat{r}(x,y)
\]

\[
F(x,y) = \begin{bmatrix} x \mid y \mid \frac{\partial x}{\partial y} \mid \frac{\partial y}{\partial x} \end{bmatrix}^\top
\]

\[
C_N = \frac{1}{n-1} \sum_{k=0}^{n-1} (x_k - \mu)(x_k - \mu)^T
\]
RegCov Smoothing - Formulation

\[ I = S + T \]

\[ S(p) = \frac{1}{2g} \sum_{q \in N(p,r)} w_{pq} I(q) \]

- Structure-texture decomposition via smoothing
- Smoothing as weighted averaging
- Different kernels (\(w_{pq}\)) result in different types of filters.
- Three novel patch-based kernels for structure texture decomposition.


Model 1

- Depends on sigma-points representation of covariance matrices (Hong et al.,CVPR’09)
  \[ C = LL^T \quad \text{Cholesky Decomposition} \]
  \[ S = \{ s_i \} \quad \text{Sigma Points} \]

\[ s_i = \begin{cases} \alpha \sqrt{d} L_i & \text{if } 1 \leq i \leq d \\ -\alpha \sqrt{d} L_i & \text{if } d+1 \leq i \leq 2d \end{cases} \]

Final representation

\[ \Psi(C) = (\mu, s_1, \ldots, s_d, s_{d+1}, \ldots, s_{2d})^T \]

Resulting kernel function

\[ w_{pq} \propto \exp \left( -\frac{\|\Psi(C_p) - \Psi(C_q)\|^2}{2\sigma^2} \right) \]

Model 2

- An alternative way is to use statistical similarity measures.
- A Mahalanobis-like distance measure to compare image patches.

\[ d(p, q) = \sqrt{(\mu_p - \mu_q)(C_p^{-1}(\mu_p - \mu_q))^T} \]

\[ C = C_p + C_q \]

Resulting kernel

\[ w_{pq} \propto \exp \left( -\frac{d(p, q)^2}{2\sigma^2} \right) \]

Model 3

resulted from a discussion with Rahul Narain (Berkeley University)

- We use Kullback-Leibler(KL)-Divergence measure from probability theory.
- A KL-Divergence form is used to calculate statistical distance between two multivariate normal distribution

\[ d_{KL}(p, q) = \frac{1}{2} \left( \mu_p^T C_p^{-1} \mu_p + \mu_q^T C_q^{-1} \mu_q - \mu_p^T C_p^{-1} \mu_q + \ln \left( \frac{\det C_p}{\det C_q} \right) \right) \]

Resulting kernel

\[ w_{pq} \propto \frac{d_{KL}(p, q)}{2\sigma^2} \]
Our filtering kernels consider local image geometry in the calculation of filtering weights by capturing texture information.

\[ \begin{align*}
12 \quad 1, \\
(x, y) & \text{ denotes the pixel location. Hence, the covariance descriptor of an image patch is computed as a 7×7 matrix. Including (x, y) into the feature set is important since it allows us to encode the correlation of other features with the spatial coordinates.} \\
& \text{The feature set can be extended to include other features, like for example rotationally invariant forms of the derivatives, if desired.} \\
& \text{In the experiments, we handle color images by computing the patch similarity weights } w_{pq} \text{ using the intensity information and taking the weighted average over the corresponding RGB vectors rather than the intensity values in Equation 23. We empirically found that including RGB components to the feature set does not change the results much but increases the running times.}
\end{align*} \]

**3.3. Model 1**

Using the set \( S \) defined by Equation (21), a vectorial representation of a covariance matrix can be obtained by simply concatenating the elements of \( S \). Moreover, first-order statistics can be easily incorporated to this representation scheme by including the mean vector of the

TV
Rudin et al. 1992

BLF
1998

WLS
Farbman et al. 2008

Envelope Extraction
Subr et al. 2009
Buades et al. 2010

Xu et al. 2011

RTV
Xu et al. 2012

Model 1
Shading preserved
Structure preserved
No unintuitive edge
Preserves shade and structure
### WLS & Local Extrema

- **Input**: Image of a sculpture.
- **WLS**
- **Local Extrema**

### RTV & Model2

- **RTV**
- **Model2**

### Multiscale Decomposition

- **$S_1(k = 5)$**
- **$S_2(k = 7)$**

---

**Input**: Image of a sculpture.

**WLS**

**Local Extrema**

**RTV**

**Model2**

**Multiscale decomposition**

- **$S_1(k = 5)$**
- **$S_2(k = 7)$**
Edge Detection

Canny edges of original image

Canny edges of smoothed image
Image Abstraction

Detail Boosting
Image Retargeting
Where we are going

- Linear filtering
- Nonlinear filtering (unified formulations)
- Pixels to Patches (context is more important than content)
- New patch representations may reveal new smoothing behaviors
- Better the smoothing, better the applications!
- Clearly, we have a long way to go to solve the problem of image smoothing!