Adaptive Manifolds for Real-Time High-Dimensional Filtering
Milestones and Advances in Image Analysis

Markus Schwinn

November 27, 2012

Motivation

\[ g_i = \frac{\sum_{p_j \in S} \phi(\hat{p}_i - \hat{p}_j) \cdot f_j}{\sum_{p_j \in S} \phi(\hat{p}_i - \hat{p}_j)} \]

- "framework" for some high-dimensional filter
- \( \phi \) is a Gaussian kernel
- possible filters for 2D colour images:
  - convolution \( \rightarrow \hat{p} \in \mathbb{R}^2 \)
  - bilateral \( \rightarrow \hat{p} \in \mathbb{R}^5 \)
  - non-local mean \( \rightarrow \hat{p} \in \mathbb{R}^{3n^2+2} \)
    - \( n \) depends on window size
- powerful... \textit{but very slow}
1. What are adaptive manifolds?

2. How to construct them?

3. What’s the use of?

Manifolds Are Not Unknown!

- practical use of manifolds
  → projecting world onto a map
Meaning of Adaptive

- approximation of input signal in high-dimensional space
- approximately linear w.r.t local neighbourhood

![Graph showing adaptive manifolds](image)

- For 2D colour image
  → dealing with 5-dimensional space
- Construction of one point on manifold
  → \( P(S_x, S_y, R_S, G_S, B_S) \)
  \( S \) denotes point in spatial domain

Computing Adaptive Manifolds

1. low-pass filtering input signal
   → generates first manifold \( \eta_1 \)

2. compute colour deviation of the pixels
   depending on manifold and original image

more technical:
largest eigenvector \( v_1 \) of

\[
(f_1 - \eta_1) \cdot (f_1 - \eta_1)^T
\]

→ \( v_1 \) describes variation of colour values
3 cluster pixels in two subsets.

- depending on "main colour"
  - defining above and below w.r.t first manifold

more technical:

\[
\text{sign} = v_1^T (f_i - \eta_{1i})
\]

\[C_+ \leftarrow p_i \text{ if sign} \geq 0\]

\[C_- \leftarrow p_i \text{ if sign} < 0\]

4 compute for each cluster manifolds \(\eta_+\) and \(\eta_-\)

- higher weighting for pixels, not represented well in \(\eta_1\)

5 repeat up from Step 2

- until number of manifolds is reached
The Algorithm

**Splatting**

projects colour for each position onto each manifold
Gaussian weighted with

\[ \Psi_{splat}(\hat{\eta}_{ki}) = \phi(\eta_{ki} - f_i)f_i \]

\( \phi \) is a Gaussian kernel
**Blurring**

Blurs over all manifolds

\[ \Psi_{splat}(\hat{\eta}_{ki}) \xrightarrow{} \Psi_{blur}(\hat{\eta}_{ki}) \]

changes information between sample points \( \eta_{ki} \)

---

**Slicing**

compute filter response

interpolates by blurred values over all adapted manifolds
The Algorithm - all fits together

The Algorithm

\[ g_i = \frac{\sum_{p_j \in S} \phi(\hat{p}_i - \hat{p}_j) \cdot f_j}{\sum_{p_j \in S} \phi(\hat{p}_i - \hat{p}_j)} \]

\[ \Rightarrow g_i = \frac{\sum_{k=1}^{K} \phi(\hat{p}_i - \hat{p}_j) \cdot \Psi_{\text{blur}}(\hat{\eta}_{ki})}{\sum_{k=1}^{K} \phi(\hat{p}_i - \hat{p}_j)} \]

\[ \Rightarrow g_i = \frac{\sum_{k=1}^{K} \phi(\eta_{ki} - f_i) \cdot \Psi_{\text{blur}}(\hat{\eta}_{ki})}{\sum_{k=1}^{K} \phi(\eta_{ki} - f_i)} \]

\[ \Rightarrow g_i = \frac{\sum_{k=1}^{K} \phi(\eta_{ki} - f_i) \cdot \Psi_{\text{blur}}(\hat{\eta}_{ki})}{\sum_{k=1}^{K} \phi(\eta_{ki} - f_i) \Psi_{\text{blur}}(\phi(\eta_{ki} - f_i))} \]

Number of Manifolds

- independent of number of pixels and dimension
- no general mechanism, sensitive to the problem
- but: depends of standard deviation of spatial- and range domain
Runtime
- clustering $\rightarrow O(dN \log K)$
- computing manifolds $\rightarrow O(dNK)$
- performing filter $\rightarrow O(dNK + dNK)$
- in total: $O(dNK)$ with $K = \text{const} \Rightarrow O(dN)$

$\Rightarrow$ high performance for runtime and good storage allocation
Edge-Aware Smoothing (5-D)
Full-HD 1920x1080 at 0.007 sec per frame

Detail Enhancement (5-D)  Input Video
Full-HD 1920x1080 at 0.007 sec per frame
Denoising with Additional Information

- add additional channels for more information
- adding an infrared channel → improving result

![noisy image](image1)
![infrared image](image2)

Denoising with Additional Information

- denoised results

![without IR channel](image3)
![with IR channel](image4)
Advantages:
- adaptable for a "general framework"
- runtime linear in number of pixels and dimension
- euclidean and also geodesic filters adaptable

Drawbacks:
- sensitive to number of manifolds
- choose of Gaussian kernels (standard deviation)

References:

Figures:
2. http://earth.imagico.de/maps/earth_large.jpg
Thank you for your Attention