Reducing Image Noise Using Spline Smoothing

Yuya Horiuchi
ECE 557 Engineering Data Analysis & Modeling
Outline

- Objective
- Introduction
- Experiment Methods
- Results
- Comparison to General Methods
- Conclusion
Objective

- To propose a new method used to reduce noise.
- To reduce noise without blurring edges.
Introduction

Spline Smoothing Criterion function

\[ E = \alpha \sum_{i=1}^{n} W_i (y_i - \hat{g}(x_i))^2 + (1-\alpha) \int_{-\infty}^{+\infty} (\hat{g}(x)''')^2 \, dx \]

- \( \alpha \) : Tradeoff value
- \( W_i \) : Weight coefficients of \( i \) th data

- Minimize \( E \)
- \( \alpha \rightarrow 1 \) : Least square linear model
- \( \alpha \rightarrow 0 \) : Cubic interpolation
Introduction

$ f = \sum (M \times N) $ of 2-dimensional data

$ (M + N) $ criterion functions

$ (M + N) $ tradeoff values

$ 2(M \times N) $ weight coefficients

How do we decide these values?

$f_{i,j} = $ intensity of point $(i, j)$
Experiment Method

Add zero mean white gaussian noise with 0.05 variance.
Experiment Method

- Repeat the following process.
  - apply the spline smoothing to the noisy image.
  - Obtain the gradient matrix by the Prewitt matrix.
  - Decide new $\alpha$'s and $W_i$'s for each line and pixel using the gradient matrix for the next step.

- Decide the initial values.
  - Let all $\alpha$'s be $\alpha_{init}$ ( $\alpha_{init}$ is arbitrary. ex. 0.5).
  - Let all $W_i$'s be 1 (uniform weight).
Experiment Method

- Prewitt matrix \( \mathbf{z} = \begin{pmatrix} -0.25 & 0 & 0.25 \\ -0.25 & 0 & 0.25 \\ -0.25 & 0 & 0.25 \end{pmatrix} \)

- Gradient Matrix \( \mathbf{G} = \mathbf{Z} \ast f \)

- New \( \alpha' \)'s and \( W_i' \)'s for the next step

\[
\alpha_i' = C \left( \max_{i} \left( \sum_{k=1}^{M} |G_{i+1,k+1}| \right) \right)
\]

\[
W_{i,j}' = W_{i,j} + \left( \left| G_{i+1,j+1} \right| - \frac{1}{M} \sum_{k=1}^{M} \left| G_{i+1,k+1} \right| \right)
\]
Results

Dependence of Initial Tradeoff value $\alpha_{\text{init}}$

![Graphs showing the dependence of initial tradeoff value $\alpha_{\text{init}}$ on mean square error and number of steps.](image)
Results - Output Images

Original noisy image

1st step

\( C = 0.9 \)

\( \alpha_V \)

\( \alpha_H \)
Results - Output Images

1st step

2nd step
Results - Output Images

2nd step

Final step

α_V

α_H
Comparison to General Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>MSE X 10^{-3}</th>
<th>MSE of grad. X 10^{-3}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spline smoothing</td>
<td>1.8089</td>
<td>1.1942</td>
</tr>
<tr>
<td>Wiener filter</td>
<td>2.0941</td>
<td>1.1208</td>
</tr>
<tr>
<td>Median filter (3 X 3)</td>
<td>2.7194</td>
<td>1.6587</td>
</tr>
</tbody>
</table>
Conclusion

- The final value of MSE is less dependent on the initial tradeoff value.
- This method could minimize MSE, but it is not as sharp as the image generated by Wiener filter.
- The method should be improved to reduce the running cost.
Truncated SVD (of pseudo inverse filter)

\[ f_{\text{tsvd}} = \sum_{i=1}^{k} \frac{u_i^* f}{\sigma_i} v_i \]

Tikhonov

\[ f_{\text{tikhonov}} = \sum_{i=1}^{N} \frac{\sigma_i^2}{\sigma_i^2 + \alpha^2} \frac{u_i^* f}{\sigma_i} v_i \]

Wiener filter

\[ f_{\text{wiener}} = \sum_{i=1}^{N} \frac{\delta_i \sigma_i^2}{\delta_i \sigma_i^2 + \lambda_i} \frac{u_i^* f}{\sigma_i} v_i \]