Image Filter Using with Gaussian Curvature and Total Variation Model

Deepak Kumar Gour, Sanjay Kumar Sharma

1,2 Dept. of Electronics & communication Engineering, UIT Bhopal, India

Abstract

Priors play an essential role in Bayesian theory in image processing. Geometric priors are very popular because of their physical explanation. The neighborhood structure of pixel can be described more accurately by its curvature. In this paper, we deals with the image restoration algorithms based on Gaussian curvature and total variation modal to achieve smooth denoising preserving the details of image. There we show that how priors should be imposed for certain types of surfaces and how they can be imposed efficiently in a variational framework. We first show a novel method that can reconstruct a closed surface from a finite point cloud then by using total variation modal we suppress the noise. We can directly extract prior information from the reconstructed surfaces. We provide parametric model and analyze its properties. The new modal can preserve more details while suppress noise. The experimental results are given to demonstrate the performance of the proposed method.

Keywords

Bayesian Theory, Gaussian Curvature (GC), Total Variation (TV) Modal, Denoising, Variational Framework and Priors

I. Introduction

Digital image processing covers many aspects. For example, image basic linear transformation and filtering, image restoration, image encoding and compression and, image reconstruction, digital image, intelligent processing, and morphological processing and so on. Estimating the image from observed data is a fundamental task in digital image processing. During the past few decades, denoising has been a hot field in image processing and numerous schemes have been proposed for the task, such as variational methods, wavelet theory, dictionary learning, compressed sensing, etc.

Among these theories, Bayesian Theorem can be used to derive variational methods. There the data $I(x)$, where $x \in \Omega \subset \mathbb{R}^2$ is the spatial coordinates, $\Omega$ is the sampling domain, and $n$ is the dimension, now we want to estimate the signal $U(x)$. This can be done by Bayesian Theorem:

$$P(U|I) = \frac{P(I|U)P(U)}{P(I)} \propto P(I|U)P(U)$$

(1)

Where $p(*)$ is the probability. By maximizing the probability $p(U|I)$ can minimize following energy

$$E(U) = -\log(P(U|I))$$

(2)

Put it in the Eq. (1), we have

$$E(U) = -\log(P(U|I)) - \log(P(U))$$

(3)

Therefore, this sampling model (derivating I from U) and some prior of U have to be assumed in this framework. In general, this links Bayesian Theorem and the variational framework:

$$\max \; \log(P(U|I)) = \log(P(U)) + \log(P(U))$$

$$\min \; E(U) = E_{\mu}(U, I) + E_{\nu}(U),$$

Where the double headed arrows indicate the counterparts in these two approaches.

The $E_{\mu}$ is the data fitting energy and the $E_{\nu}$ is the regularization energy. $E_{\mu}$ indicates how well an estimation of $U$ fits the data $I$. This depends on the sampling process. The $\ell_2$ norm is commonly used because the measurement error usually satisfies Gaussian distribution, which leads to $E_{\mu} = \frac{|I - U|^2}{\sigma^2}$ ($\sigma$ is a parameter). Another frequent choice is the $\ell_1$ norm which corresponds to Laplacian noise model. In most of models, $E_{\nu}$ is a regularization term that imposes the prior knowledge of the $U$, such as Tikhonov, the $\ell_2$ norm of the gradient, symmetry, gradient distribution [5, 7], Total Variation (TV) [1-2, 5], Mean Curvature (MC) [5], or Gaussian Curvature (GC) [3-6].

Among these regularizations, GC and TV are interesting because we have observed that GC filter is better in edge-preserving and TV filter is better in removing noise. Several researchers have shown that the properties of GC [4-5, 9] and TV [1-2, 5, 9]. In this paper, we show that GC and TV filter can be impose together with some changes in their minimal projection.

A. Traditional Solvers

Traditional solvers for variational models, such as gradient decent method, Split Bregman Iteration and Primal Dual methods, usually require computing the gradient of the total energy functional. Requiring the total energy to be differentiable makes imposing arbitrary imaging models (noise, blurring, inpainting, supper resolution, scatter light, etc.) difficult.

These types of methods are suffered from the numerical stability requirement. Therefore, the step size in each iteration is limited. They usually need a large number of iterations to converge. Because of that traditional solvers are usually memory intensive, which makes them computational expensive. They require the system memory to be at least several times larger than the input image size. This leads to an issue for large images, where required memory for traditional solvers may be larger than the system memory.

B. Motivation

Besides the complicate equation, large memory requirements, another drawback of traditional methods is that they are computationally slow. The main reason is that these previous approaches start from the total energy $E(U)$ in Eq. (4) without considering the geometric meaning of minimizing curvature. Another problem for previous solvers is that they are not generic. To accelerate the computation, the multigrid strategy is introduced.
With the multigrid acceleration, the GC and TV are faster than traditional methods. From these computational methods and filters we have found that the GC filter use in image processing for smoothing and denoising the image. GC filter is better in preserving details i.e. GC filter is better in edge-preserving but reduces its ability in noise suppression. And TV filter is better in removing noise but it also removes the details of the image.

C. Contribution
To overcome these issues, we have proposed the combination of GC and TV, with some modification in these two frameworks and using these two variational models. We have adopted the property of these two variational frameworks and formed a new filter. We have found some better performance as shown in the experiment section. There to minimize the regularization energy. Our method is inspired by the observation that regularization energy $E_{\Phi}$ is the dominant part during the minimization. As shown in Fig. 1, the regularization energy $E_{\Phi}$ usually decreases while the energy $E_{\phi}$ usually increases if the initial condition is the original image. Since the total energy has to decrease, $E_{\Phi}$ must be the dominant part. Therefore, as long as the decreased amount in $E_{\Phi}$ is larger than the increased amount in $E_{\phi}$, the total energy $E$ decreases. There are several benefits of doing so. First, we do not require the total energy to be differential. Therefore, it can handle arbitrary complex noise model. Second, the edges are preserved. Third, the resulting filter is simple to compute, and its physical meaning is clear.

![Fig. 1: Regularization is the Dominant Part in Optimization](image)

II. Gaussian Curvature and Total Variation Regularization
First, we show the mathematical form of Gaussian Curvature and Total Variation regularization

A. Gaussian Curvature
Let $x \in \Omega$ denote the spatial coordinates and $I(i,j): x \rightarrow R^+$ denote the given discrete digital image with coordinates $i$ and $j$. Let $U(x)$ denote the unknown signal, i.e., the desired output image to be estimated. We interpret the signal as a geometric surface over the space of the data, i.e., $\psi = (x,U(x))GC$ is defined as

$$K(U(x)) = \frac{U_xU_{xx} - U_x^2}{(1+U_x^2+U_y^2)^2}$$

Recall that the total GC $K$ of any surface is related to the surface’s topology through the Gauss-Bonnet theorem:

$$\int_{\partial \Omega} K d\gamma + \int_{\Omega} K d\Omega = 2\pi\chi(\Omega)$$

Where $K_\partial$ is the boundary curvature, $d\gamma$ a length element, and $\chi$ the Euler characteristic of $\Omega$. Since total GC is a topological invariant, one can only minimize total absolute GC [5, 9]. Surfaces with zero total absolute GC are called developable. The total absolute GC variational model is

$$\min_{U \in L^2} \left\{ E(U) = \int_\Omega |K(U)| d\gamma \right\},$$

s.t. $\int_\Omega |U-I|^2 dx < \epsilon$,

where $E(U)$ is the GC energy and $\epsilon$ is a given termination threshold. $L^2$ is the space of square-integrable functions.

B. Total Variation
We can use a similar approach to construct a filter to solve TV models. We use the constrained ROF model [1, 5]

$$E(U) = \int_\Omega |\nabla U(x)| dx,$$

s.t. $\|U(x) - I(x)\| \leq \lambda$

Our filter solver is summarized in Algorithms, which define the GC and TV filter. In the local projection operator $P, \delta P$, respectively, we directly use the piecewise constancy assumption.

III. Domain Decomposition
Locally minimizing the smaller absolute principal curvature is prevented by dependencies between neighboring pixels. We introduce here a domain decomposition algorithm to defasance these dependencies.

We decompose the discrete domain of an image $U$ into two disjoint subsets, the “white” points $W$ and the “black” points $B$. We further split each of these two subsets: white triangles $W_T$, white circles $W_C$, black triangles $B_T$, and black circles $B_C$. This decomposition guarantees that neighbors are in different subsets, as illustrated in Fig. 2.

![Fig. 2 : Illustration of Disjoint Domain Decomposition](image)

This decomposition has several benefits: First, it removes the dependencies between neighboring pixels. For example, when BC needs to be updated, all pixels in BC can be updated simultaneously. Second, this is independence, the update can use the neighbors that have already been updated. This guarantees convergence.
Third, in a $3 \times 3$ local window, all tangent planes $TS$ can be formed. Therefore, proximal projection can be used to make the surface $U(x)$ more developable, which means locally reducing the Gaussian curvature. We need to project $U(x)$ to $\bar{U}(x)$ such that $\bar{U}(x)$ is on the closest tangent plane of a neighboring pixel.

Enumeration of all Projections In order to find the tangent plane in $N(x)$ that has the smallest $\kappa_1$, we find all possible tangent in a $3 \times 3$ pixel neighborhood of $x$ that do not include $x$ as a vertex. There are in total 12 such tangents: six through each of the four white neighbors $W$, six through the four black neighbors.

Since some of the 12 tangent triangles share common edges over $x$, and projecting onto these edges is sufficient, there are only 12 different $d_i$, six to the common edges from $W$, six to the common edges from $B$.

**IV. Algorithm Implementation**

In this filter, we use all possible linear forms (that are minimal projection) to approximate the data and choose the minimal change to update the current estimation. Since the minimal projection of both new GC and TV is used and then it use to update the pixels, this filter is more efficient in minimizing principle curvature than GC and TV filter separately, as shown in the experiment section.

**A. Minimal Projection Operator by modified Gauss operator**

After computing all $\{d_i, i = 1, \ldots, 12\}$ we use the smallest absolute distance as the minimum projection of the current intensity $U(x)$ to the target intensity $\bar{U}(x)$ such that $\bar{U}(x)$ is on one of the tangent planes through the neighboring pixels. More specifically, we find $d_m$ such that $|d_m|_{\text{min}} = \min \{d_i, i = 1, \ldots, 12\}$. Then, we let $\bar{U}(x) = U(x) + d_m$. We denote this local update operation by $P_{\text{new}}$. It needs minimum operations (plus, minus, divide). This operator, as summarized in Algorithm 1, is thus compact and efficient.

The set $\{d_i, i = 1, \ldots, 12\}$ is a complete description of the local geometry at $x$. For any given $U(x)$ and its $\kappa_1 = \min \{\kappa_i, i = 1, \ldots, 12\}, U(N(x))$ can be obtained by solving a linear system of equations in Algorithm 1. Moreover, $d_i$ is the linear curvature in the corresponding direction. Based on the Euler Theorem, we have $d_i \approx K_1 \cos^2 \theta + K_2 \cos^2 \theta$, where $K_1, K_2$ are the principle curvatures and $\theta$ is the angle to the principle plane. Therefore, if the angular sampling $\hat{\theta}$ is dense enough in $[-\pi, \pi]$, we have $\kappa_1 = \min \{\kappa_1\}$ when $K_1, K_2 \geq 0$. We use $d_m$ as an approximation of the minimal absolute principle Curvature.

**B. New Gauss Curvature Operator**

We iterate $P_{\text{new}}$ over all pixels in each of $B_T, B_C, W_T$, and $W_C$. Since the pixels within each set are independent, the iteration order does not matter. This yields our curvature filter, as summarized in Algorithms. It is clear that has linear computational complexity with respect to the total number of pixels.

Because of the domain decomposition, all pixels in the same set are independent of each other and the projection can be applied in parallel. This enables us to prove convergence of Algorithm, and also accelerates convergence since each update is based on already updated neighbors.

Algorithm 1 Projection Operator $P_{\text{new}}$

Require: $U(i, j)$

1. $d_1 = (U_{i-1,j} + U_{i+1,j}) / 2 - U_{i,j}$
2. $d_2 = (U_{i,j-1} + U_{i,j+1}) / 2 - U_{i,j}$
3. $d_3 = (U_{i-1,j-1} + U_{i+1,j+1}) / 2 - U_{i,j}$
4. $d_4 = (U_{i-1,j+1} + U_{i+1,j-1}) / 2 - U_{i,j}$
5. $d_5 = (U_{i+1,j} + U_{i,j-1}) / 2 - U_{i,j}$
6. $d_6 = (U_{i+1,j+1} + U_{i,j-1}) / 2 - U_{i,j}$
7. $d_7 = (U_{i,j+1} + U_{i,j-1}) / 2 - U_{i,j}$
8. $d_8 = (U_{i+1,j} + U_{i,j+1}) / 2 - U_{i,j}$
9. $d_9 = (U_{i+1,j+1} + U_{i,j+1}) / 2 - U_{i,j}$
10. $d_{10} = (U_{i,j+1} + U_{i,j+1}) / 2 - U_{i,j}$
11. $d_{11} = (U_{i+1,j} + U_{i+1,j}) / 2 - U_{i,j}$
12. $d_{12} = (U_{i+1,j+1} + U_{i+1,j+1}) / 2 - U_{i,j}$

$\text{find } d_m, \text{ that } |d_m| = \min \{d_i, i = 1, \ldots, 12\}$

Ensure: $U(i, j) = U(i, j) + d_m$

Algorithm II New Filter $G_{\text{new}}$

Require: $U(i, j)$

1. $\forall x \in BT$, $P_g(U(x))$
2. $\forall x \in BC$, $P_g(U(x))$
3. $\forall x \in WT$, $\bar{P}_g(U(x))$
4. $\forall x \in WC$, $\bar{P}_g(U(x))$
5. Ensure: $U(i, j)$

**C. Proposed Modal**

According to the previous discussion, as described above the Gaussian Curvature filter and Total Variation filter have some merits and demerits. In this paper, from their properties, we have picked up their merits as and apply them together to find better result. There we have modified the gauss curvature operator as described above in algorithm I, after that algorithm we have applied the algorithm of total variation modal [9]. With that, we have found improved performance, as shown in the experiment section.

**V. Experiments**

In this section, we discuss the results of our numerical experiments to define the effectiveness of the proposed method. In this
experiment we select the 512x512 pixels of the Lena standard test image. It was degraded with Salt and Pepper Noise of 5, 10 and 30 % density. As Shown in Fig. 5 and Table 1.

For the comparison of performance and to verify the effectiveness and reliability of the recovery of the proposed filter, we performed on PC (Intel(R) Core(TM)2 @ 3.00GHz, 2.0 GB memory) with Matlab simulation software 2015b algorithm programming. We investigated Gaussian Curvature (GC) and Total Variation (TV) filter. It is well-known that the GC filter is better in preserving details, especially for the low value of noise and that TV filter is better in removing noise, it can remove both lower and higher valued noise. For a fair comparison we have used same number of iterations i.e. 60.

To evaluate performance of the noise detection, we have used three level of noise density. The results of noise detection for salt and pepper noise are shown in the table 1 and also the experimental output shown in fig. 5. You see in the table and fig.5 (Second row), the performance of the Gaussian Curvature filter worsened at higher level of noise. And also see in fig. 5 (third row) and table the quality filtered image with total variation filter is worsened. Now the proposed modal of filter is better in preserving edges as well as better in removing noise as compared with Total Variation (TV) or Gaussian Curvature (GC). As shown in Fig. 5 (bottom row) and table1.

The energy profiles of Gaussian Curvature filter, Total Variation filter and proposed filter are shown in the fig. 6 That indicate the regularization energy of the all given filters and their comparison. The fig. 4 shows the performance and ability of the proposed filter.

<table>
<thead>
<tr>
<th>Noise Density (%)</th>
<th>SNR</th>
<th>PSNR</th>
<th>SSIM</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salt &amp; pepper Noise</td>
<td>5%</td>
<td>12.7933</td>
<td>18.4497</td>
<td>0.3302</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>9.8007</td>
<td>15.4571</td>
<td>0.1732</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>5.0221</td>
<td>10.6785</td>
<td>0.0523</td>
</tr>
<tr>
<td>GC Filter</td>
<td>5%</td>
<td>26.1305</td>
<td>31.7869</td>
<td>0.9121</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>24.7095</td>
<td>30.3658</td>
<td>0.8940</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>19.4697</td>
<td>25.1261</td>
<td>0.7822</td>
</tr>
<tr>
<td>TV Filter</td>
<td>5%</td>
<td>23.0798</td>
<td>28.7361</td>
<td>0.8698</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>22.3671</td>
<td>28.0234</td>
<td>0.8268</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>20.7965</td>
<td>25.6703</td>
<td>0.7550</td>
</tr>
</tbody>
</table>

Fig. 4: (a) Original Image (left), (b) Filtered Image (Mid.), (c) Difference between these original Image and filtered image

Table 1: Results and Comparison in SNR, PSNR, SSIM, MSE of the Image Denoising Experiment

(a) 5% Noise Density, (b) 10% Noise density, (c) 30% Noise density

Fig. 5: Comparison of the Results With a Real Image Filtered by GC Filter (Second Row), TV Filter (Third Row) and Proposed Filter (Bottom Row) from Noisy Images Having Different Noise Density (Top Row), Column Wise From Top to Bottom

(a)
VI. Conclusion

We have proposed new algorithm for image denoising based on Gaussian curvature and total variation modal. It is a fast novel filter that is use the same assumptions as the respective variational models, they can minimize the regularization energy for the corresponding variational models. The proposed filter is guarantees that not only image edges are are well preserved or other small-scaled structures but also remove the noise effectively at high level of noise. The filter is much faster in both runtime and convergence. The filter is parameter-free and easy to implement and parallelize. The experiment results of deferent method are given.

In future, this filter solver can be parallelized to obtained higher performance, to increase the speed of the filtration and to perform on the large images. This filter can also be combined with wavelet transform.

References


