# Robust Scale-Space Filter Using Second-Order Partial Differential Equations

Bumsub Ham, *Student Member, IEEE,* Dongbo Min, *Member, IEEE*, and Kwanghoon Sohn, *Member, IEEE*

***Abstract*— This paper describes a robust scale-space filter that adaptively changes the amount of flux according to the local topology of the neighborhood. In a manner similar to modeling heat or temperature flow in physics, the robust scale-space filter is derived by coupling Fick's law with a generalized continuity equation in which the source or sink is modeled via a specific heat capacity. The filter plays an essential part in two aspects. First, an evolution step size is adaptively scaled according to the local structure, enabling the proposed filter to be numerically stable. Second, the influence of outliers is reduced by adaptively compensating for the incoming flux. We show that classical diffusion methods represent special cases of the proposed filter. By analyzing the stability condition of the proposed filter, we also verify that its evolution step size in an explicit scheme is larger than that of the diffusion methods. The proposed filter also satisfies the maximum principle in the same manner as the diffusion. Our experimental results show that the proposed filter is less sensitive to the evolution step size, as well as more robust to various outliers, such as Gaussian noise, impulsive noise, or a combination of the two.**

***Index Terms*— Anisotropic diffusion, edge preserving filter, heat equation, scale-space, thermal diffusivity.**

## I. Introduction

REAL WORLD objects can be decomposed into different structures and scales according to underlying tenets of scale-space theory introduced by Witkin [1]. Scale-space consists of a 3-D volume, made up of a set of 2-D planes, where signals are represented by different scale parameters. Many researchers in the signal processing community have gone to great lengths to obtain semantically meaningful 3-D volumes and to understand the structures that are inherently embedded in the signals. Scale-space representations can be classified into two categories according to the strategies used to construct a scale-space: isotropic and anisotropic scale-space.

Witkin constructed an isotropic scale-space by convolving an original signal with a Gaussian kernel [1]. Koenderink proved that a structure of an isotropic scale-space is governed by the heat equation (or diffusion equation), describing a distribution of heat or temperature on a given domain over time [2]. The resolution of the signal becomes coarse as time goes on, or as the variance of the Gaussian kernel increases, and vice versa. He presented two criteria for the scale-space: *causality* and *homogeneity and isotropy*. *Causality* means that any feature at a coarse resolution is required to possess a cause at a finer resolution, which is equivalent to the maximum principle from a parabolic partial differential equation (PDE) [3]. *Homogeneity and isotropy* means that the kernel is space-invariant, which leads to a poor localization at the coarse resolution, where features such as edges and boundaries are not directly available. To localize these features accurately, tracking across the scale space is needed, which was proved to be a complex task [4].

Perona and Malik changed the scale space paradigm from isotropic to anisotropic by altering a thermal diffusivity in the heat equation from a constant to a space-variant function called the "edge-stopping" function. Therefore, the *homogeneity and isotropy* criterion was replaced with *immediate localization* and *piecewise smoothing* criteria [5]. *Piecewise smoothing* means that intra-region smoothing is preferred to inter-region smoothing at all scales. *Immediate localization*, which is a result of *piecewise smoothing*, signifies that object boundaries should be sharp, coinciding with the meaningful boundaries at each resolution. You *et al*. analyzed the behavior of the Perona and Malik model with the shape of an energy surface [6]. They formulated anisotropic diffusion (AD) as an optimization problem, and solved it by finding a minimum of a given energy function via the steepest descent method. AD meets the three criteria well and constructs the scale-space exactly, but it is sensitive to outliers. A comprehensive review can be found in [7].

A number of methods have been proposed to alleviate the influence of outliers. Catte *et al*. resolved the problem by introducing a regularized space-variant function [8]. The smoothed gradient used for calculating the thermal diffusivity makes AD more stable in the presence of noise, while guaranteeing its existence and uniqueness [8]. Black *et al*. posed AD as an energy minimization problem with robust statistics [9]. They showed that the "edge-stopping" function in AD is closely linked to an error norm and an influence function in the robust estimation framework, and then presented a new "edge-stopping" function based on Tukey's biweight robust estimator [10], [11]. Carmona and Zhong proposed an adaptive AD that controls the amount of diffusion, as well as the direction of diffusion by decomposing the gradient into the normal and tangential components [12]. This enables the noise in

Manuscript received June 14, 2011; revised January 31, 2012; accepted May 5, 2012. Date of publication May 24, 2012; date of current version August 22, 2012. This work was supported by the National Research Foundation of Korea Grant funded by the Korea Government (MEST) under Grant 2012-0004995. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Joseph P. Havlicek.

B. Ham and K. Sohn are with the School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea (e-mail: mimo@yonsei.ac.kr; khsohn@yonsei.ac.kr).

D. Min is with the Advanced Digital Sciences Center, 138632 Singapore (e-mail: dongbo@adsc.com.sg).

Digital Object Identifier 10.1109/TIP.2012.2201163

the features to be eliminated without damaging them. It is similar to a tensor based diffusion framework [13], such as edge enhancing diffusion [14], coherence enhancing diffusion [15], and autocorrelation-driven diffusion [16]. Recently, Tschumperle and Deriche [17] proposed a unified diffusion PDE scheme by demonstrating the relationships among functional minimization [6], [9], divergence expressions [2], [5], [8], and a framework of oriented Laplacians [12].

These methods, however, have only tried to mitigate the effect of an additive white Gaussian noise (AWGN), so they cannot handle other outliers such as impulsive or speckle noise. You and Scott proposed a speckle reducing AD tailored to ultrasonic and radar imaging applications [18] that proved to be an anisotropic counterpart to the conventional adaptive speckle filter. Ling and Bovik proposed a hybrid filter for de-noising molecular images by incorporating a median filter into AD [19]. Each filter is operated alternatively so that the impulsive noise is eliminated, but it lacks a theoretical background.

In this paper, we add a *robustness* criterion to the scale-space, and propose a robust scale-space filter, which adaptively changes the amount of flux according to the local topology of the neighborhood. In a similar manner to modeling heat or temperature flow in physics, the proposed filter is derived from coupling Fick's law with a generalized continuity equation, in which the source or sink is modeled via a specific heat capacity. It is shown that classical diffusion methods, such as isotropic diffusion and AD, are special cases of the proposed filter. By analyzing the stability condition of the proposed filter, we also verify that its evolution step size in an explicit scheme is larger than that of the diffusion methods.

This paper is organized as follows Section II reviews the isotropic and ADs. Section III describes the robust scale-space filter with a numerical analysis. We then describe properties of the proposed filter, including the stability and maximum principle, in Section IV. Analysis of the experimental results is presented in Section V. Finally, Section VI concludes this paper with some suggestions for future work.

## II. Scale Space Paradigm

In this section, we first present the generalized second-order PDE model used to construct the scale-space filter, and briefly review conventional scale-space filterings: isotropic and ADs.

### A. Generalized Second-Order PDE

Let $I(x,y,t)$, $n(x,y,t)$, $c(x,y,t)$: $\Omega_C \times \mathrm{T}_C \rightarrow \mathbb{R}^+$ be a function with a continuous image domain $\Omega_C = \{(x,y)|\, 0 \le x \le X, 0 \le y \le Y\} \subset \mathbb{R}^2$ and a continuous time domain $\mathrm{T}_C = \{t|\, 0 \le t < \infty\} \subset \mathbb{R}$. The generalized second-order PDE used to construct the scale-space filter is as follows [5]:

$$n(x,y,t)\partial_t I(x,y,t) - c(x,y,t)\Delta I(x,y,t) - \nabla c(x,y,t) \cdot \nabla I(x,y,t) = 0 \quad (1)$$

$$I(x,y,0) = U(x,y) \quad (2)$$

$$\partial_n I(x,y,t) = 0 \quad (3)$$

where $\partial_t$ and $\partial_n$ represent a partial derivative with respect to $t$, and a directional derivative in a normal direction. $\Delta$ and $\nabla$ denote the Laplace and gradient operator, respectively. $I(x,y,t)$ denotes a 3-D volume composed of 2-D continuous images $I(x,y)$, according to the scale or time $t$ with an initial condition $U(x,y)$. $c(x,y,t)$ represents the thermal diffusivity at time $t$ (specifically, the thermal conductivity), which controls the quantity of heat or temperature. $n(x,y,t)$ is an arbitrary function which can vary with respect to time $t$ and spatial coordinate $(x,y)$. It corresponds to a specific heat capacity in the proposed method, as will be seen in Section IV-A.

### B. Isotropic Diffusion

The isotropic scale-space is constructed from a well-known second-order parabolic PDE, called the heat (or diffusion) equation as follows:

$$\partial_t I(x,y,t) = c(x,y,t)\Delta I(x,y,t) = c\Delta I(x,y,t). \quad (4)$$

The constant $c$ represents the thermal diffusivity. Note that the isotropic diffusion is equivalent to the generalized second-order PDE as in (1) in which $n(x,y,t) = 1$ and $c(x,y,t) = c$. It was proved that the solution of the heat equation is theoretically equivalent to the result of the convolution integral between an initial image and a time-varying Gaussian kernel as follows [2]:

$$I(x,y,t) = \begin{cases} U(x,y), & t = 0 \\ G_{\sqrt{2t}}(x,y) * U(x,y), & t > 0 \end{cases} \quad (5)$$

where $G_\sigma(x,y)$ denotes the Gaussian kernel with standard deviation $\sigma$. Note that (3) becomes a solution when a constant $c$ is set to 1.

It also meets the *causality* criterion in a given domain $\Omega_C \times \mathrm{T}_C$, as shown in [7]

$$\inf_{\Omega_C} U(x,y) \le I(x,y,t) \le \sup_{\Omega_C} U(x,y). \quad (6)$$

The isotropic diffusion is efficiently implemented by separating the 2-D Gaussian kernel into two 1-D Gaussian kernels. However, it has ambilaterality. Since a space-invariant kernel is used, it smooths both noise and features evenly regardless of a local structure; therefore, it dislocates features at the coarse resolution.

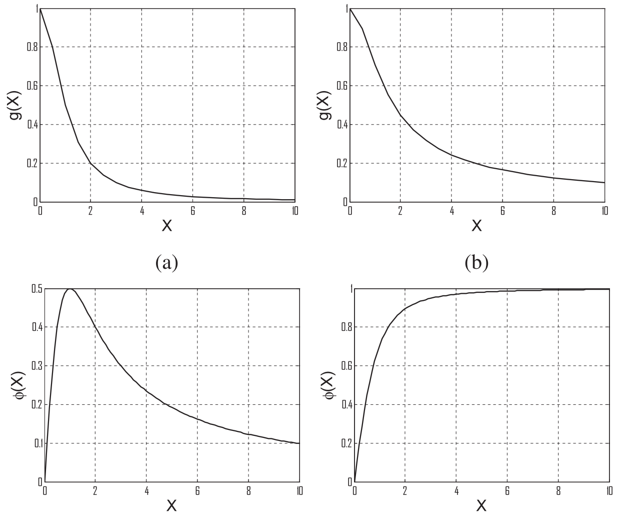
### C. AD

Perona and Malik resolved the feature dislocation problem by incorporating prior knowledge into the isotropic diffusion, leading to AD [5]. They introduced a time and spatially varying diffusivity function into the diffusion model. It is posed as an initial value problem with an adiabatic boundary condition such that

$$\partial_t I(x,y,t) = \nabla \cdot [c(x,y,t)\nabla I(x,y,t)] \quad (7)$$

where $\nabla\cdot$ represents the divergence operator. The thermal diffusivity function is as follows:

$$c(x,y,t) = g\left(\|\nabla I(x,y,t)\|\right). \quad (8)$$



(a) (b)

(c) (d)

Fig. 1. Diffusivity functions $g(X)$ of the AD. (a) and (b) Diffusivity functions in (11) and (12) when $\lambda$ is set to 1, whose potential functions are nonconvex and convex, respectively. The corresponding flux functions $\Phi(X)$ are plotted in (c) and (d), respectively.

The gradient magnitude is denoted by $\|\nabla I(x,y,t)\|$. $g(X)$ is an "edge-stopping" function that satisfies $g(X) \to 0$ as $X \to \infty$, stopping diffusion across different features. Note that any function satisfying this criterion can be used as an "edge-stopping" function. Note also that the generalized second-order PDE in (1) becomes AD as in (7) when $n(x,y,t) = 1$ and $c(x,y,t) = g(\|\nabla I(x,y,t)\|)$.

It is well-known that (7) is derived by minimizing the following energy functional via the gradient descent method [6], [7], [9]:

$$E(I) = \int_{\Omega_C} \Psi\left(\|\nabla I(x,y,t)\|\right) d\Omega_C \tag{9}$$

where $\Psi(X)$ denotes a potential function whose gradient is represented by a flux function $\Phi(X)$

$$\begin{aligned}\nabla\Psi\left(\|\nabla I(x,y,t)\|\right) &= \Phi\left(\nabla I(x,y,t)\right)\\ &= g\left(\|\nabla I(x,y,t)\|\right)\nabla I(x,y,t).\end{aligned} \tag{10}$$

Perona and Malik used the following "edge-stopping" function:

$$g(X) = \frac{1}{1 + X^2/\lambda^2} \tag{11}$$

where $\lambda$ is a bandwidth parameter used in the noise estimator. Fig. 1(a) shows this function when $\lambda$ is set to 1. When $|X| \geq \lambda$, the potential function become nonconvex, so an inverse diffusion occurs. As shown in Fig. 1(c), when $\Phi'(X) < 0$, the slope of the edge increases as the evolution (time) goes on, which results in an enhancement of features. See [5], [8] for more information. Charbonnier *et al.* [10] used the "edge-stopping" function as follows:

$$g(X) = \frac{1}{\sqrt{1 + X^2/\lambda^2}}. \tag{12}$$

Fig. 1(b) shows this function when $\lambda$ is set to 1. This function always makes the potential function convex, so the flux function monotonically increases regardless of $\lambda$, and a forward diffusion always occurs. When $\Phi'(X) > 0$ as shown in Fig. 1(d), the slope of the edge decreases as the evolution (time) goes on [5].

## III. Robust Scale-Space Filter

Although AD gives us the denoised solution for AWGN to some extent, it is not robust to an impulsive noise or a mixture noise. In this section, we derive the robust scale-space filter with a numerical discretization scheme where a new *robustness* criterion is enforced.

### A. Problem Statements

We first examine why AD is not robust to the outliers. First, it considers relations only between neighborhoods, rather than among neighborhoods; therefore, the thermal diffusivity is independently calculated between a center node and each neighborhood, which causes the flux $\Phi(X)$ to flow symmetrically. It means that AD is an adiabatic process which is not suitable to handle the outliers, for example, impulsive noise. Second, it cannot distinguish inliers with outliers unless the bandwidth parameter $\lambda$ adaptively changes according to the local statistics, for example, $\lambda$ should be set large if the outliers such as impulsive noises exist, which results in a poor localization. To address these problems, we judge outliers and then compensate for the incoming flux by considering local characteristics of the neighborhood.

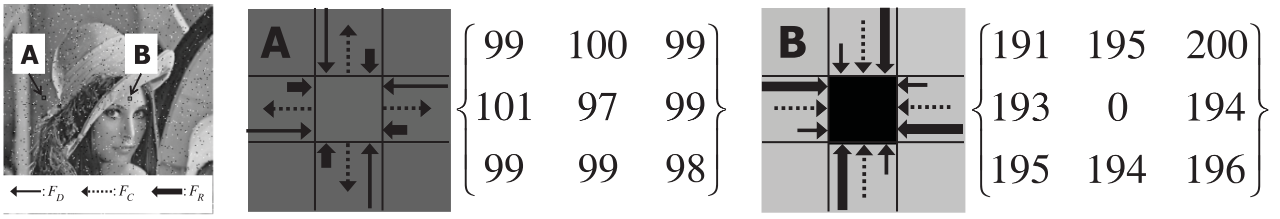
Let us consider an example shown in Fig. 2. The solid arrow indicates the flux of AD $F_D$, and the dotted arrow indicates the additional flux $F_C$ by the proposed filter. An incoming flux of the proposed filter $F_R$, denoted by the blocked arrow, is defined as $F_D \pm F_C$ in the log domain. In case A, the intensity value of the center node is similar to that of the neighborhood nodes, so the incoming flux $F_R$ should be smaller than the flux $F_D$ of AD for a good localization. In case B, the incoming flux $F_R$ should be larger than $F_D$, since the intensity value of the center node is extremely different from that of the neighborhood nodes, such that the center node is likely to be an outlier (caused by an impulsive noise). Note that the flux of AD almost does not flow in case B, since the thermal diffusivity is very low.

In conclusion, the divergence of the flux of the filter should decrease in case of A, and increase in case of B, in order to make the scale-space filter robust to the outliers.

### B. Derivation of a Robust Scale-Space Filter

We assume that the compensated flux adjusts the magnitude of the flux only, and does not change the direction. In other words, two fluxes $F_D$ and $F_R$ in Fig. 2 should have the same direction. Also, the compensated flux is determined by the neighboring information. We model the compensated flux with the sum of the diffusivity, since it can be utilized as an indicator of the outliers, for example, when the sum of the diffusivity is small as in case B of Fig. 2, it implies that the center node is likely to be an outlier.

Based on this assumption, a robust scale-space filter is derived via the coupling of Fick's law with a generalized



$$\begin{Bmatrix} 99 & 100 & 99 \\ 101 & 97 & 99 \\ 99 & 99 & 98 \end{Bmatrix}$$

$$\begin{Bmatrix} 191 & 195 & 200 \\ 193 & 0 & 194 \\ 195 & 194 & 196 \end{Bmatrix}$$

(a) (b) (c)

Fig. 2. Example of flux compensation. (a) Noisy image. (b) Diffusion at piecewise smooth region. (c) Diffusion at outliers corrupted by impulsive noise. In order to handle an outlier such as (a), flux is adaptively compensated according to the characteristics of the neighborhood. The solid arrow indicates the flux by AD $F_D$, and the dotted arrow indicates the additional flux $F_C$. An incoming flux of the proposed filter $F_R$ denoted by the blocked-arrow is defined as $F_D \pm F_C$ in the log domain. (b) In case A, the intensity value of the center node is similar to that of neighborhood nodes, so for a good localization, the incoming flux $F_R$ should be smaller than flux $F_D$ of the AD. (c) In case B, the incoming flux $F_R$ should be larger than $F_D$, since the center node is likely to be an outlier.

continuity equation. Fick's law states that the concentration gradient $\nabla I(x, y, t)$ generates a diffusion flux $J$, which aims to compensate for a concentration field such that

$$J = -c(x, y, t)\nabla I(x, y, t). \tag{13}$$

The thermal diffusivity (thermal conductivity) is represented as the function $g(X)$ as follows:

$$c(x, y, t) = g\left(\|\nabla I(x, y, t)\|\right). \tag{14}$$

Similar to AD, in the robust scale-space filter, any function satisfying the "edge-stopping" criterion, for example, (11) or (12), can be used as the function $g(X)$.

The general form of the continuity equation is as follows:

$$\partial_t I(x, y, t) = -\nabla \cdot J + s(x, y, t) \tag{15}$$

where $s(x, y, t)$ is a function that describes the generation or removal of $I$, called a source or sink. It generalizes an advection equation that is related to the Navier–Stokes equation in fluid dynamics [20]. Note that the coupling of Fick's law with the continuity equation where $s(x, y, t)$ is set to 0 becomes the isotropic diffusion as in (4), or AD as in (7), according to the "edge-stopping" function $c(x, y, t)$. Subsequently, this makes the diffusion transport a fixed quantity of the flux without destroying or creating it. They cannot, however, handle outliers such as a mixture of Gaussian noise and an impulsive noise.

Alternatively, the source or sink exists in the robust scale-space filter, allowing it to model the compensated flux according to the local structure. Even though the proposed filter does not conserve the total energy, the existence of the source or sink makes it more robust to various outliers, as will be described in the next section.

Plugging Fick's law into the general continuity equation, it is induced as follows:

$$\begin{aligned} \partial_t I(x, y, t) &= -\nabla \cdot J + s(x, y, t) \\ &= \nabla \cdot [c(x, y, t)\nabla I(x, y, t)] + s(x, y, t). \end{aligned} \tag{16}$$

We re-formulate the above equation using a new function $\kappa$

$$\partial_t I(x, y, t) = \kappa \nabla \cdot [c(x, y, t)\nabla I(x, y, t)] \tag{17}$$

where

$$\kappa = \frac{\nabla \cdot [c(x, y, t)\nabla I(x, y, t)] + s(x, y, t)}{\nabla \cdot [c(x, y, t)\nabla I(x, y, t)]}. \tag{18}$$

As mentioned previously, an additional flux function $\kappa$ adjusts the magnitude of the flux only while preserving its direction, which always makes $\kappa$ positive. Thus, without loss of generality, we analyze this equation in the log domain by assuming that an equilibrium state is reached as time approaches infinity

$$\ln\left[\partial_t I(x, y, t)\right] = \ln\left[\nabla \cdot (c(x, y, t)\nabla I(x, y, t))\right] + \ln(\kappa). \tag{19}$$

$\ln(\kappa)$ can be interpreted in two aspects. First, it represents the quantity of the divergence of the compensated flux $F_C$ in Fig. 2. Second, it acts as a source or sink according to its sign. Recall that, in the robust scale-space filter, the compensated flux $\kappa$ is modeled with the sum of the diffusivity, which makes the divergence of the flux decrease when the sum of the diffusivity is high and vice versa. We can model this quantity with an additional flux function as follows:

$$\ln(\kappa) = -\ln\left[\left|\nabla \cdot (c(x, y, t)\nabla A(x, y, t))\right|\right] \tag{20}$$

where $A(x, y, t)$ is an *auxiliary* function which guides the sum of the thermal diffusivity. The divergence quantity of the compensated flux $|\nabla \cdot (c(x, y, t)\nabla A(x, y, t))|$ can be utilized as an indicator of the outlier, so finally determines the velocity of the diffusion, taking into account the presence of outliers. Note that the divergence of additional flux takes the absolute value to meet the constraint of $\kappa$, that is, it adjusts the magnitude of the flux only while preserving its direction.

In this paper, we assume that the function $A(x, y, t)$ meets the following conditions. First, it has a unit gradient, therefore the diffusivity values of neighbors are just summed without scaling. Second, it is periodic in order to make an absolute value of the gradient of $A(x, y, t)$ equal to 1 regardless of position $(x, y)$ and time $t$. Truly, $\nabla \cdot (c(x, y, t)\nabla A(x, y, t))$ is anisotropic, since the diffusivity function $c(x, y, t)$ manages the diffusion process using the gradient magnitude $\|\nabla I(x, y, t)\|$. Actually, it can be seen that the function $A(x, y, t)$ in the compensated flux $c(x, y, t)\nabla A(x, y, t)$ corresponds to the image $I(x, y, t)$ in ADs flux $c(x, y, t)\nabla I(x, y, t)$. $A(x, y, t)$ plays a role only in guiding the sum of the diffusivity, so is called an *auxiliary* function. Any function satisfying these criteria can be used, which will be presented in the next section.

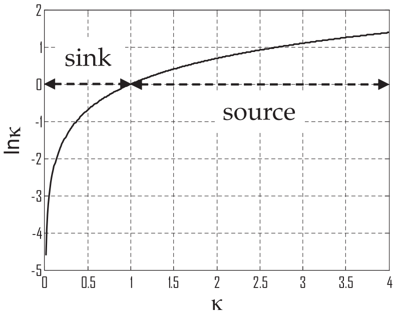


Fig. 3. Divergence quantity of flux. The neighborhood node becomes a "sink" when $\ln\kappa < 0$, and a "source" otherwise. When $\ln\kappa = 0$, the neighborhood node becomes neither a "sink" nor a "source," so the divergence quantity of flux is maintained.

Plugging (20) into (19) leads to

$$ln\,[\partial_t I(x,y,t)] = ln\,[\nabla\cdot(c(x,y,t)\nabla I(x,y,t))] - ln\,[|\nabla\cdot(c(x,y,t)\nabla A(x,y,t))|]. \quad (21)$$

Equation (21) can be classified into three cases according to a sign of $ln\,(\kappa)$ as follows.

1) $\ln(\kappa) < 0$ ($|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]| > 1$, $s(x,y,t) < 0$): it occurs when the center node is similar to the neighborhood node. The neighborhood node becomes a "sink," as shown in Fig. 2(b), which results in a decrease in the divergence quantity of the flux in the robust scale-space filter.
2) $\ln(\kappa) > 0$ ($|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]| < 1$, $s(x,y,t) > 0$): this occurs when the center node is extremely different from the neighborhood node. The neighborhood node becomes the "source," as shown in Fig. 2(c), which results in increasing the divergence quantity of the flux in the robust scale-space filter.
3) $\ln(\kappa) = 0$ ($|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]| = 1$, $s(x,y,t) = 0$): the robust scale-space filter becomes AD since $s(x,y,t)$ is 0; therefore, the neighborhood node becomes neither a "sink" nor a "source," so the divergence quantity of the flux is maintained. Note that this case occurs only when $\kappa$ is equal to 1.

Fig. 3 shows that the divergence quantity of the flux depends on the value of $\kappa$. After eliminating a log function in (21), it becomes as follows:

$$\partial_t I(x,y,t) = \frac{\nabla\cdot[c(x,y,t)\nabla I(x,y,t)]}{|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]|}. \quad (22)$$

Note that (22) can also be induced by setting $n(x,y,t)$ and $c(x,y,t)$ in (1) to $|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]|$ and $g\,(\|\nabla I(x,y,t)\|)$, respectively

$$|\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]|\,\partial_t I(x,y,t) - \nabla\cdot[c(x,y,t)\nabla I(x,y,t)] = 0. \quad (23)$$

The robust scale-space filter is therefore obtained by finding a solution of a generalized second-order PDE where a source or sink exists, as in (15).

### C. Implementation

We discretize (22) with an explicit finite difference scheme. This scheme guarantees numerical stability for only a small range of evolution step size, but we will show that the robust scale-space filter with the explicit scheme has a relatively larger range of an evolution step size than that of the conventional diffusion.

Let $I(i,j,t)$, $n(i,j,t)$, $c(i,j,t)$: $\Omega_D \times \mathrm{T}_D \rightarrow \mathbb{R}^+$ be functions with a 2-D discrete domain $\Omega_D = \{(i,j)|\,0 \le i \le M, 0 \le j \le N\} \subset \mathbb{N}^2$ and a discrete time domain $\mathrm{T}_D = \{t|\,0 \le t < \infty\} \subset \mathbb{N}$. $I_{i,j}^t$ denotes the intensity of $(i,j)$ when time is equal to $t$.

Equation (22) is equivalent to

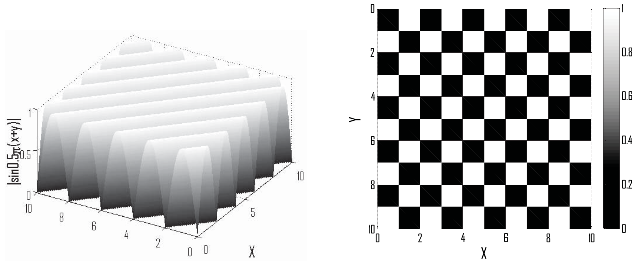
$$\partial_t I(x,y,t) = \frac{\partial_x\,[c(x,y,t)\partial_x I(x,y,t)] + \partial_y\left[c(x,y,t)\partial_y I(x,y,t)\right]}{\left|\partial_x\,[c(x,y,t)\partial_x A(x,y,t)] + \partial_y\left[c(x,y,t)\partial_y A(x,y,t)\right]\right|}. \quad (24)$$

First, we consider a discretized approximation of $\partial_x\,[c(x,y,t)\partial_x A(x,y,t)]$ via forward and backward differences as follows:

$$\begin{aligned}
&\partial_x\,[c(x,y,t)\partial_x A(x,y,t)]\\
&\approx \frac{1}{2}\partial_x\left[c(x,y,t)\frac{1}{\Delta i}(A_{i,j}^t - A_{i-\Delta i,j}^t)\right]\\
&\quad + \frac{1}{2}\partial_x\left[c(x,y,t)\frac{1}{\Delta i}(A_{i+\Delta i,j}^t - A_{i,j}^t)\right]\\
&\approx \frac{1}{2}\frac{1}{(\Delta i)^2}\left[c_{i+\Delta i,j}^t(A_{i+\Delta i,j}^t - A_{i,j}^t)\right]\\
&\quad - \frac{1}{2}\frac{1}{(\Delta i)^2}\left[c_{i,j}^t(A_{i,j}^t - A_{i-\Delta i,j}^t)\right]\\
&\quad + \frac{1}{2}\frac{1}{(\Delta i)^2}\left[c_{i,j}^t(A_{i+\Delta i,j}^t - A_{i,j}^t)\right]\\
&\quad - \frac{1}{2}\frac{1}{(\Delta i)^2}\left[c_{i-\Delta i,j}^t(A_{i,j}^t - A_{i-\Delta i,j}^t)\right]\\
&= \frac{1}{2}\frac{1}{(\Delta i)^2}\left[\begin{array}{l}(c_{i+\Delta i,j}^t + c_{i,j}^t)(A_{i+\Delta i,j}^t - A_{i,j}^t)\\ +(c_{i-\Delta i,j}^t + c_{i,j}^t)(A_{i-\Delta i,j}^t - A_{i,j}^t)\end{array}\right]. \quad (25)
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
&\partial_y\,(c(x,y,t)\partial_y A(x,y,t))\\
&\approx \frac{1}{2}\frac{1}{(\Delta j)^2}\left[\begin{array}{l}(c_{i,j+\Delta j}^t + c_{i,j}^t)(A_{i,j+\Delta j}^t - A_{i,j}^t)\\ +(c_{i,j-\Delta j}^t + c_{i,j}^t)(A_{i,j-\Delta j}^t - A_{i,j}^t)\end{array}\right] \quad (26)\\
&\partial_x\,(c(x,y,t)\partial_x I(x,y,t))\\
&\approx \frac{1}{2}\frac{1}{(\Delta i)^2}\left[\begin{array}{l}(c_{i+\Delta i,j}^t + c_{i,j}^t)(I_{i+\Delta i,j}^t - I_{i,j}^t)\\ +(c_{i-\Delta i,j}^t + c_{i,j}^t)(I_{i-\Delta i,j}^t - I_{i,j}^t)\end{array}\right] \quad (27)\\
&\partial_y\,(c(x,y,t)\partial_y I(x,y,t))\\
&\approx \frac{1}{2}\frac{1}{(\Delta j)^2}\left[\begin{array}{l}(c_{i,j+\Delta j}^t + c_{i,j}^t)(I_{i,j+\Delta j}^t - I_{i,j}^t)\\ +(c_{i,j-\Delta j}^t + c_{i,j}^t)(I_{i,j-\Delta j}^t - I_{i,j}^t)\end{array}\right]. \quad (28)
\end{aligned}$$



(a) (b)

Fig. 4. (a) Example of an auxiliary function $A(x, y, t)$ as in (35) and (b) its discretization. Since $A(x, y, t)$ is an auxiliary function which guides the sum of the diffusivities, it is periodic and has unit gradient. The origin is the most upper left pixel. There are two cases according to the mod of $(i + j)$. Let $(i + m, j + n)$ be a 4-neighboring pixel of $(i, j)$. Then, $A^t_{i+m,j+n} - A^t_{i,j}$ is 1 when $(i + j)$ is even, and $-1$ otherwise.

Now, we consider a discretized approximation of $(c^t_{i+\Delta i,j} + c^t_{i,j})$, which can be referred to as a projection of the gradient along one direction [5], such that

$$\begin{aligned}&\frac{1}{2}(c^t_{i+\Delta i,j} + c^t_{i,j})\\ &\quad\approx \frac{1}{2}\left[g\left(\left|I^t_{i+\Delta i,j} - I^t_{i,j}\right|\right) + g\left(\left|I^t_{i+\Delta i,j} - I^t_{i,j}\right|\right)\right]\\ &\quad= g\left(\left|I^t_{i+\Delta i,j} - I^t_{i,j}\right|\right).\end{aligned} \tag{29}$$

Note that the first and second terms are approximated by the backward and forward differences, respectively.

Similarly, we have

$$\frac{1}{2}(c^t_{i-\Delta i,j} + c^t_{i,j}) \approx g\left(\left|I^t_{i-\Delta i,j} - I^t_{i,j}\right|\right) \tag{30}$$

$$\frac{1}{2}(c^t_{i,j+\Delta j} + c^t_{i,j}) \approx g\left(\left|I^t_{i,j+\Delta j} - I^t_{i,j}\right|\right) \tag{31}$$

$$\frac{1}{2}(c^t_{i,j-\Delta j} + c^t_{i,j}) \approx g\left(\left|I^t_{i,j-\Delta j} - I^t_{i,j}\right|\right). \tag{32}$$

Plugging all equations [from (25) to (32)] into (24) and letting $\Delta i = \Delta j = 1$ leads to

$$\begin{aligned}\partial_t I(x, y, t) \approx &\begin{bmatrix} g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right)\left(I^t_{i+1,j} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\left(I^t_{i-1,j} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right)\left(I^t_{i,j+1} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right)\left(I^t_{i,j-1} - I^t_{i,j}\right)\end{bmatrix}\\ &\cdot\left[\left|\begin{matrix} g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right)\left(A^t_{i+1,j} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\left(A^t_{i-1,j} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right)\left(A^t_{i,j+1} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right)\left(A^t_{i,j-1} - A^t_{i,j}\right)\end{matrix}\right|\right]^{-1}.\end{aligned} \tag{33}$$

As mentioned earlier, $A(x, y, t)$ is an *auxiliary* function that guides the sums of the diffusivity. This function has a unit gradient, $|A^t_{i+1,j} - A^t_{i,j}|$, $|A^t_{i-1,j} - A^t_{i,j}|$, $|A^t_{i,j+1} - A^t_{i,j}|$, and $|A^t_{i,j-1} - A^t_{i,j}|$ equal to 1, which aggregates the diffusivity without scaling them. Furthermore, an absolute value of the gradient of $A(x, y, t)$ equal to 1 regardless of position $(x, y)$ and time $t$, since the function $A(x, y, t)$ is periodic. One example follows such that:

$$A(x, y, t) \equiv A(x, y) = 1 - \left|\cos\left(\frac{\pi}{2}(x + y)\right)\right|. \tag{34}$$

Fig. 4(a) shows another example satisfying these criteria, which is written as follows:

$$A(x, y, t) \equiv A(x, y) = \left|\sin\left(\frac{\pi}{2}(x + y)\right)\right|. \tag{35}$$

Fig. 4(b) shows a discrete version of $A(x, y, t)$. The origin is the most upper left pixel. There are two cases according to the mod of $(i + j)$. Let $(i + m, j + n)$ be a 4-neighboring pixel of $(i, j)$. Then, $A^t_{i+m,j+n} - A^t_{i,j}$ is 1 when $(i + j)$ is even, $-1$ otherwise. Therefore, the denominator in (33) can be induced as follows:

$$\begin{aligned}&\left|\begin{matrix} g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right)\left(A^t_{i+1,j} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\left(A^t_{i-1,j} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right)\left(A^t_{i,j+1} - A^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right)\left(A^t_{i,j-1} - A^t_{i,j}\right)\end{matrix}\right|\\ &\quad= g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right) + g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\\ &\quad\quad+ g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right) + g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right).\end{aligned} \tag{36}$$

Plugging (36) into (33) and approximating the partial derivative with respect to the time, we obtain the final equation

$$\begin{aligned}I^{t+1}_{i,j} = I^t_{i,j} + \tau &\begin{bmatrix} g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right)\left(I^t_{i+1,j} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\left(I^t_{i-1,j} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right)\left(I^t_{i,j+1} - I^t_{i,j}\right)\\ +g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right)\left(I^t_{i,j-1} - I^t_{i,j}\right)\end{bmatrix}\\ &\cdot\begin{bmatrix} g\left(\left|I^t_{i+1,j} - I^t_{i,j}\right|\right) + g\left(\left|I^t_{i-1,j} - I^t_{i,j}\right|\right)\\ +g\left(\left|I^t_{i,j+1} - I^t_{i,j}\right|\right) + g\left(\left|I^t_{i,j-1} - I^t_{i,j}\right|\right)\end{bmatrix}^{-1}\end{aligned} \tag{37}$$

where $\tau$ is an evolution step size. For simplifying the notation, it can be represented as follows:

$$I^{t+1}_{i,j} = I^t_{i,j} + \tau\frac{\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)\left(I^t_{k,l} - I^t_{i,j}\right)}{\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)} \tag{38}$$

where $N(i, j) = \{(i - 1, j), (i + 1, j), (i, j - 1), (i, j + 1)\}$ represents the 4-neighborhood of the center node, $I^t_{i,j}$.

Note that (38) is similar to an implementation of AD [9], except for the normalization term that makes the proposed filter more robust to the outliers. Although the right term in (38) looks similar to a bilateral filter [21], the proposed method shows a different behavior, which will be discussed in Section IV-D.

## IV. Properties of a Robust Scale-Space Filter

### A. Specific Heat Capacity

The normalization term $|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|$ is closely related to the specific heat capacity. The thermal diffusivity in the heat equation is defined as the thermal conductivity divided by the volumetric heat capacity as follows [22]:

$$\upsilon = \frac{k}{\rho \cdot c_p} \tag{39}$$

where $k$ and $\rho$ represent the thermal conductivity and density, respectively. The specific heat capacity is represented as $c_p$. This means that materials with high thermal diffusivity adapt their temperature to that of their neighborhood *rapidly*, and vice versa. This behavior partially coincides with the AD. That is, diffusion rarely occurs by the "edge-stopping" function when the center node has a different distribution of its neighborhood. The thermal diffusivity in AD, however, does not include the specific heat capacity, so rapidity is not taken into account. The thermal diffusivity in AD is modeled as the thermal conductivity when both the density and specific heat capacity are set to 1. In this case, the thermal conductivity $k$ in the thermal diffusivity implicitly coincides with $g(\nabla I)$ in AD.

The specific heat capacity per mass is defined as $c_p = \partial C/\partial m$, where $C$ and $m$ represent the heat capacity and mass, respectively. In the absence of a phase transition, $c_p = C/m$. Substances with a high specific heat capacity have a low thermal diffusivity $\upsilon$. Therefore, the materials with a high specific heat capacity $c_p$ slowly adapt their temperature to that of their neighborhood, and vice versa.

The proposed method can also be viewed from a different perspective

$$\begin{aligned} I_{i,j}^{t+1} &= I_{i,j}^{t} + \tau \frac{1}{\sum_{(k,l)\in N(i,j)} g\left(\left|I_{k,l}^{t} - I_{i,j}^{t}\right|\right)} \\ &\quad \cdot \sum_{(k,l)\in N(i,j)} g\left(\left|I_{k,l}^{t} - I_{i,j}^{t}\right|\right)\left(I_{k,l}^{t} - I_{i,j}^{t}\right). \end{aligned} \tag{40}$$

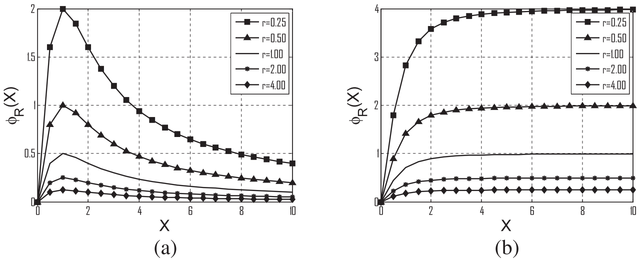
This exactly includes the specific heat capacity in the thermal diffusivity $c_p$, that is, the normalization term exactly coincides with $c_p$. Generally, the pure material has a higher specific heat capacity than the mixture. In the case of an image, the normalization term becomes large in homogeneous regions (pure materials), leading to "*slow diffusion*," whereas it becomes low in noisy regions (blended materials), leading to "*fast diffusion*." Therefore, the thermal conductivity $k$, which is equivalent to $g(\nabla I)$, implicitly represents the diffusion direction and strength, while the specific heat capacity $c_p$, which is equivalent to $\sum g(\nabla I)$, represents the diffusion velocity.

In conclusion, the robust scale-space filter adaptively changes the thermal conductivity $k$ as well as the specific heat capacity $c_p$ in the thermal diffusivity according to the "*property*" between neighborhoods and the "*concentration*" among neighborhoods. Therefore, it better reflects the physical phenomenon of diffusion smoothing.

### B. Flux

The thermal diffusivity in the robust scale-space filter is defined as follows:

$$D(x, y, t) = \frac{c(x, y, t)}{|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|}. \tag{41}$$



(a) (b)

Fig. 5. Flux function of a robust scale-space filter when (a) diffusivity function of (11) and (b) diffusivity function of (12) are used. The flux function $\Phi_R(X)$ of the robust scale-space filter is defined as $\Phi(X)/r = g(X)X/r$, where $r = |\nabla \cdot [g(\|\nabla I\|)\nabla A]|$ is the sum of the divergence of the additional flux $F_C$ in Fig. 2, when $r$ is varying from 0.25 to 4.0. Since $0 < g(X) \leq 1$, $r$ has a value between 0 and 4 when a 4-neighborhood is used.

The normalization term $|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|$ changes the magnitude of $D(x, y, t)$ according to the properties of neighborhoods. Note that $D(x, y, t)$ becomes the thermal diffusivity $c(x, y, t)$ of the classic AD, if $|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|$ is fixed to 1. The flux $\Phi_R$ of the robust scale-space filter is then defined as follows:

$$\begin{aligned} \Phi_R(\nabla I(x, y, t)) &= D(x, y, t)\nabla I(x, y, t) \\ &= \frac{\Phi(\nabla I(x, y, t))}{|\nabla \cdot [g(\|\nabla I(x, y, t)\|)\nabla A(x, y, t)]|} \\ &\approx \frac{g\left(\left|I_{k,l}^{t} - I_{i,j}^{t}\right|\right)\left(I_{k,l}^{t} - I_{i,j}^{t}\right)}{\sum_{(k,l)\in N(i,j)} g\left(\left|I_{k,l}^{t} - I_{i,j}^{t}\right|\right)} \end{aligned} \tag{42}$$

where $\Phi(\nabla I(x, y, t))$ represents the flux of AD $F_D$ in Fig. 2.

In contrast to AD, the flux function of the robust scale-space filter varies according to the relationships *between* neighborhoods, $c(x, y, t)$, and *among* neighborhoods, $|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|$. Therefore, either of the two should be fixed for observing the behavior of the flow of the robust scale-space filter. Let us assume that the property of the neighborhood is specified, such that $|\nabla \cdot [c(x, y, t)\nabla A(x, y, t)]|$ is a fixed variable with respect to a spatial coordinate $(x, y)$, not a function. This enables us to compare the flux of the robust scale-space filter with that of AD. Since $0 < g(X) \leq 1$ in (11) or (12), the sum of the diffusivity $|\nabla \cdot [g(\|\nabla I(x, y, t)\|)\nabla A(x, y, t)]|$ has a value between 0 and 4, if 4-neighbor is used as in (38). Fig. 5(a) and (b) shows the flux function $\Phi_R(X) = \Phi(X)/r = g(X)X/r$ corresponding to the diffusivity function as in (11) and (12), respectively, when $r$ is varying from 0.25 to 4.0. Note that when $r$ is 1.0, the flux function of the robust scale-space filter becomes that of AD, such that $\Phi_R(X) = \Phi(X)$.

The quantity of flux varies according to the sum of diffusivity $r = |\nabla \cdot [g(\|\nabla I(x, y, t)\|)\nabla A(x, y, t)]|$, which reflects the characteristics of the neighborhood, as well as the flux of AD $g(\|\nabla I(x, y, t)\|)\nabla I(x, y, t)$. In other words, in contrast to AD (when $r$ is always 1), the flux function of the robust scale-space filter is adaptively scaled according to the characteristics of the neighborhood for handling outliers better. As described in Fig. 2, this results in decreasing the flux when the intensity values of the neighborhood are similar (e.g., when $r$ is close to 4) and vice versa.

The forward or inverse diffusions are determined according to the convexity of the potential function used in the robust scale-space filter. Fig. 5(a) shows that if the nonconvex potential function of (11) is used, an inverse diffusion occurs when the flux starts to decrease. In contrast, if the convex potential function of (12) is used, a forward diffusion always occurs with varying quantity according to the local structure, as shown in Fig. 5(b).

### C. Stability

The stability of the numerical iteration depends on the evolution step size. Although there is no lower bound on this parameter, a small evolution step size usually leads to slow convergence. We show that the stable range of the evolution step size of a robust scale-space filter is larger than that of AD in the same explicit scheme. Let us suppose that $0 < g(|I^t_{k,l} - I^t_{i,j}|) \leq 1$, (38) can then be modified as follows:

$$I^{t+1}_{i,j} = (1-\tau)\, I^t_{i,j} + \tau \left[ \frac{\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right) I^t_{k,l}}{\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)} \right]. \tag{43}$$

Since the weight of the center node should exist between 0 and 1 in order to ensure stability, the stability condition is as follows:

$$0 \leq \tau \leq 1. \tag{44}$$

In contrast, since AD with the explicit scheme [9] is

$$\begin{aligned} I^{t+1}_{i,j} &= \left[1 - \tau \sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)\right] I^t_{i,j} \\ &\quad + \tau \left[\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right) I^t_{k,l}\right] \end{aligned} \tag{45}$$

the stability condition satisfies

$$0 \leq 1 - \tau \max\left\{\sum_{(k,l)\in N(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)\right\} \leq 1. \tag{46}$$

Therefore, its stability condition is

$$0 \leq \tau \leq \frac{1}{4}. \tag{47}$$

Similarly, if $c$ in (4) is set to 1.0, the stability condition of the isotropic diffusion is

$$0 \leq \tau \leq \frac{1}{4}. \tag{48}$$

The stability region of the robust scale-space filter is larger than that of both isotropic and AD, since it includes the "*edge-stopping*" concept in the image domain as well as the "*adaptive-evolution-step*" concept in the time domain.

### D. Robustness

As mentioned earlier, the robust scale-space filter can handle various outliers such as AWGN, impulsive noise, or a combination of the two. The right-side term in (38) is decomposed into two parts again: the nominator $\sum_{(k,l)\in N(i,j)} g(|I^t_{k,l} - I^t_{i,j}|)(I^t_{k,l} - I^t_{i,j})$ and the denominator $\sum_{(k,l)\in N(i,j)} g(|I^t_{k,l} - I^t_{i,j}|)$. The nominator term can be thought of as a weighted mean that discriminates a true signal from the AWGN, preserving features in the same manner as the AD does. Note that AD cannot handle the impulsive noise. Alternatively, the denominator term has the properties of a median filter, thus having a very low value when the impulsive noise exists, as shown in Fig. 2(c). This characteristic leads the incoming flux to increase, or the evolution step size to be largely scaled. Therefore, the robust scale-space filter implicitly embeds an impulsive noise detector, whereas most existing methods consist of two separate parts, noise detector and noise eliminator, to eliminate an impulsive noise [19], [23], [24].

It is worth noting that the robustness comes from the characteristics of both the nominator and denominator terms. Let us consider bilateral filtering [21], which also contains the normalization term as follows:

$$I^{t+1}_{i,j} = \frac{\sum_{(k,l)\in \tilde{N}(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right) I^t_{k,l}}{\sum_{(k,l)\in \tilde{N}(i,j)} g\left(\left|I^t_{k,l} - I^t_{i,j}\right|\right)}. \tag{49}$$

For fair comparison, it is assumed that the kernel size is equal to that of the robust scale-space, that is, $\tilde{N}(i,j) = (k,l) \,|(i,j), (i-1,j), (i+1,j), (i,j-1), (i,j+1)$. Note that $\tilde{N}(i,j)$ includes the center node $I^t_{i,j}$.

Let us consider a simple 1-D example as follows:

$$I^t_{i-1} \equiv \alpha,\ I^t_i \equiv \hat{\alpha}\ , I^t_{i+1} \equiv \alpha \tag{50}$$

where $\alpha$ and $\hat{\alpha}$ represent arbitrary intensity values. $\hat{\alpha}$ is an outlier corrupted by impulsive noise, therefore $\hat{\alpha} \gg \alpha$ or $\hat{\alpha} \ll \alpha$, and the ground truth intensity of $I_i$ is $\alpha$. Their corresponding weights are then defined as follows:

$$\begin{aligned} g\left(\left|I^t_{i-1} - I^t_i\right|\right) &\equiv w_{i-1} \\ g\left(\left|I^t_i - I^t_i\right|\right) &\equiv w_i \\ g\left(\left|I^t_{i+1} - I^t_i\right|\right) &\equiv w_{i+1}. \end{aligned} \tag{51}$$

Note that $w_{i-1}$ and $w_{i+1}$ are close to 0, since $g(X)$ is a monotonically decreasing function. The filtered value is then represented by

$$I^{t+1}_i = \frac{\alpha w_{i-1} + \hat{\alpha} w_i + \alpha w_{i+1}}{w_{i-1} + w_i + w_{i+1}} \approx \frac{\hat{\alpha} w_i}{w_i} = I^t_i. \tag{52}$$

Therefore, the outlier cannot be handled by the bilateral filter. AD shows a similar behavior as well

$$I^{t+1}_i = \hat{\alpha} + \tau\left[w_{i-1}\left(\alpha - \hat{\alpha}\right) + w_{i+1}\left(\alpha - \hat{\alpha}\right)\right] \approx \hat{\alpha} = I^t_i. \tag{53}$$

Alternatively, the robust scale-space filter shows a different behavior such that

$$\begin{aligned} I^{t+1}_i &= I^t_i + \tau \frac{\left[w_{i-1}\left(\alpha - \hat{\alpha}\right) + w_{i+1}\left(\alpha - \hat{\alpha}\right)\right]}{w_{i-1} + w_{i+1}} \\ &= \hat{\alpha} + \tau\left(\alpha - \hat{\alpha}\right) \\ &= (1-\tau)\,\hat{\alpha} + \tau\alpha. \end{aligned} \tag{54}$$

When $0 < \tau \leq 1$, $I^{t+1}_i$ approaches the ground truth value $\alpha$ in an iterative manner. One interesting fact is that as $\tau$ approaches 1, the convergence rate increases. Especially, when $\tau$ is set to 1, the ground truth value is obtained after one iteration only.

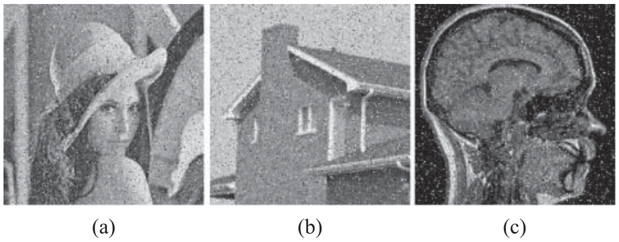


Fig. 6. Images degraded by both AWGN (noise variance = 400) and impulsive noise (density = 0.05). (a) *Lena*. (b) *House*. (c) *Brain*.

Accordingly, the denominator in the robust scale-space filter has the role of not only preventing the overall energy from diverging like bilateral filtering, but also handling various outliers better by adaptively compensating for the incoming flux.

### *E. Maximum Principle*

Including itself, let us denote the maximum and minimum of the neighborhood of $I_{i,j}^t$ as $M_{i,j}^t$ and $m_{i,j}^t$, respectively

$$M_{i,j}^t = \max\left\{I_{i,j}^t, I_{i+1,j}^t, I_{i-1,j}^t, I_{i,j+1}^t, I_{i,j-1}^t\right\} \tag{55}$$

$$m_{i,j}^t = \min\left\{I_{i,j}^t, I_{i+1,j}^t, I_{i-1,j}^t, I_{i,j+1}^t, I_{i,j-1}^t\right\}. \tag{56}$$

Define

$$\begin{aligned}(c_T)_{i,j}^t \equiv\ & g\left(\left|I_{i-1,j}^t - I_{i,j}^t\right|\right) + g\left(\left|I_{i+1,j}^t - I_{i,j}^t\right|\right) \\ & + g\left(\left|I_{i,j+1}^t - I_{i,j}^t\right|\right) + g\left(\left|I_{i,j-1}^t - I_{i,j}^t\right|\right).\end{aligned} \tag{57}$$

Assuming $0 < g(X) \leq 1$, (38) can be written as follows:

$$\begin{aligned}I_{i,j}^{t+1} &\leq M_{i,j}^t\left[1 - \frac{\tau}{(c_T)_{i,j}^t} \cdot (c_T)_{i,j}^t\right] + \frac{\tau}{(c_T)_{i,j}^t} \cdot (c_T)_{i,j}^t \cdot M_{i,j}^t \\ &= M_{i,j}^t.\end{aligned} \tag{58}$$

Similarly

$$\begin{aligned}I_{i,j}^{t+1} &\geq m_{i,j}^t\left[1 - \frac{\tau}{(c_T)_{i,j}^t} \cdot (c_T)_{i,j}^t\right] + \frac{\tau}{(c_T)_{i,j}^t} \cdot (c_T)_{i,j}^t \cdot m_{i,j}^t \\ &= m_{i,j}^t.\end{aligned} \tag{59}$$

Therefore, the discretized scheme of the robust scale-space filter in (38) satisfies the maximum principle as follows:

$$m_{i,j}^t \leq I_{i,j}^{t+1} \leq M_{i,j}^t. \tag{60}$$

This means that no maxima and minima appear at all scales. Furthermore, a continuous form of the robust scale-space filter in (23) also satisfies the maximum (or minimum) principle (see Appendix in [5]).

## V. Experimental Results

### *A. Experimental Environments*

*1) Test Sequences:* We use the following images: *Lena*, *House*, and *Brain*. They are corrupted with different types of noise: AWGN where the noise variance is 400; impulsive noise where the density is 0.05; and a mixture noise combining AWGN and the impulsive noise as shown in Fig. 6. In a subjective evaluation, we only show the results for the *Lena* image with the mixture noise due to space limitations.

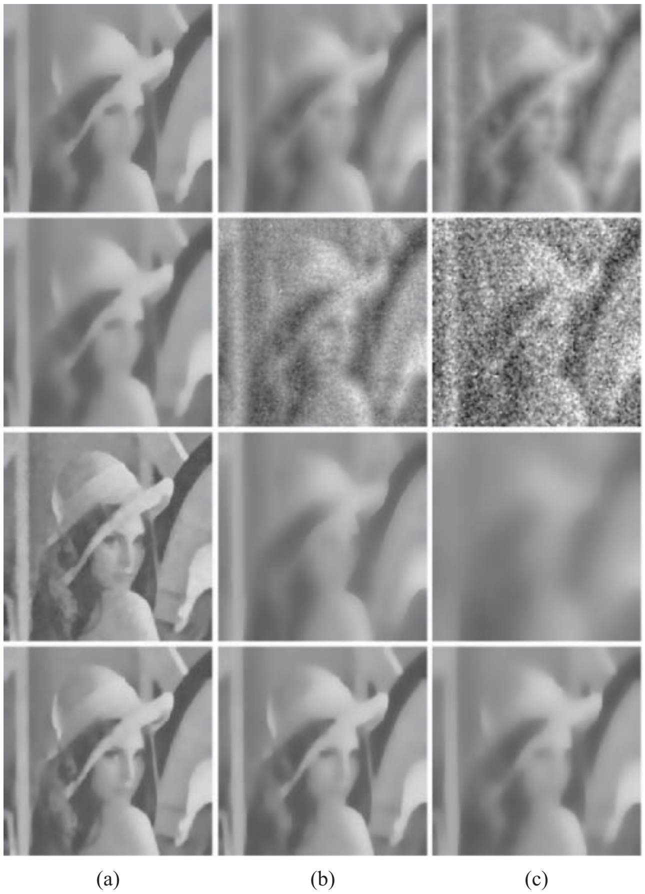


Fig. 7. Results for *Lena* image with mixture noise in Fig. 6(a) filtered by (from top to bottom) AD, regularized AD (RAD), anisotropic median diffusion (AMD), and robust scale-space filtering (RSF). An evolution step size $\tau$ is set to (a) 0.25, (b) 0.50, and (c) 1.00, respectively. The number of iterations and the bandwidth parameter $\lambda$ are set to 200 and 2.0, respectively. An edge-stopping function in (12) is used whose potential function is convex. Please see the electronic version for better visibility.

*2) Parameter Setting:* In Section V-B, the performance of the proposed method is evaluated in both a qualitative and quantitative manner. The RSF is compared with AD [5], RAD [8], and AMD [19]. For fair comparison, 4-neighborhood is used in all methods. The two "edge-stopping functions" in (11) and (12) are used, whose potential functions are nonconvex and convex, respectively. In the classic diffusion methods, it has been known that the bandwidth parameter should be set large enough to discriminate the noise from the true signal, when the impulsive noise exists. The large value, however, may cause the important features to be blurred. In contrast, the RSF has an excellent discriminative power for both Gaussian and impulsive noise even with the relatively small bandwidth parameter ($\lambda = 2.0$ in all experiments). Gaussian regularization is used in the RAD where the variance and kernel size are 1.0 and 3, respectively. The kernel size of the median filter is set to 3 in the AMD.

In Section V-C, the results of RSF with 8-neighborhood and 24-neighborhood are shown. In this case, the "edge-stopping" function in (11) with a bandwidth parameter $\lambda$ being 2.0 is used, whose potential function is nonconvex.

We observe a behavior of the bilateral filter in mixture noise environment, and compare it with RSF in Section V-D.

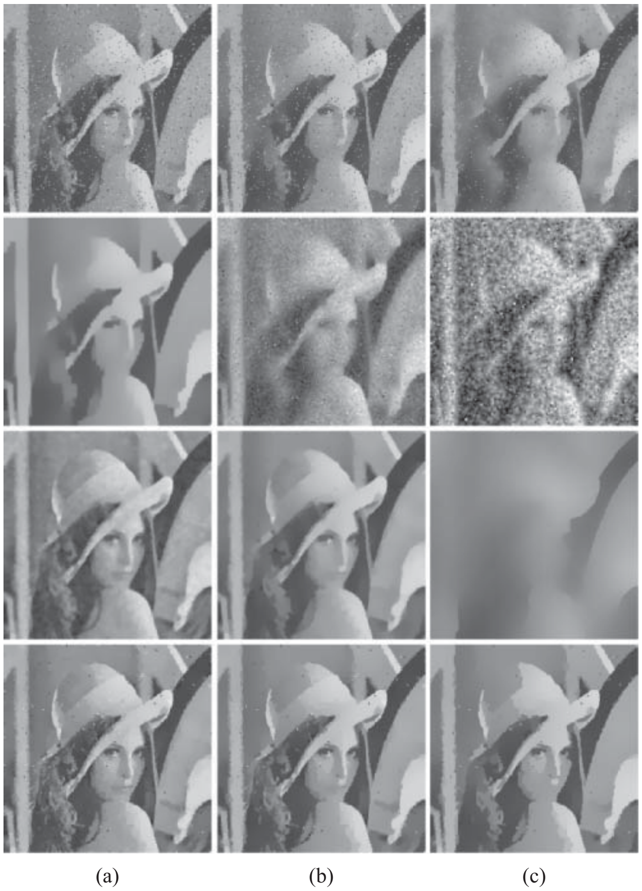
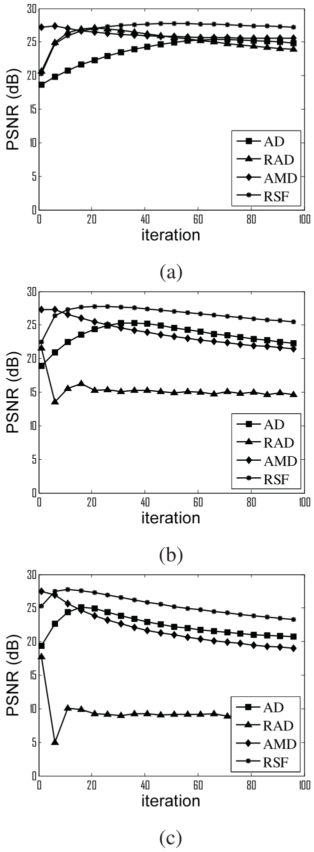


Fig. 8. Results for *Lena* image with mixture noise in Fig. 6(a) filtered by (from top to bottom) AD, RAD, AMD, and RSF. An evolution step size $\tau$ is set to (a) 0.25, (b) 0.50, and (c) 1.00, respectively. The parameters are the same as those in Fig. 7, except that the number of iterations is 500 and the edge-stopping function in (11) is used, whose potential function is nonconvex. Please see the electronic version for better visibility.



(a)

(b)

(c)

Fig. 9. Quantitative comparison for the *Lena* image corrupted by mixture noise as shown in Fig. 6(a). The peak signal to noise ratio (PSNR) is measured against a varying number of iterations and the evolution step size: (a) $\tau = 0.25$, (b) $\tau = 0.50$, and (c) $\tau = 1.00$, respectively.

The Gaussian kernel with a size $11 \times 11$ is used. The bandwidth parameter is set to the standard deviation of the AWGN, 20.

### *B. Comparison With Related Diffusion Methods*

*1) Qualitative Comparison:* Fig. 7 shows the results for the *Lena* image corrupted by the mixture noise as shown in Fig. 6(a), with an evolution step size varying at (a) 0.25, (b) 0.50, and (c) 1.00, respectively. The results were obtained by (from top to bottom) AD, RAD, AMD, and RSF with the edge stopping function in (12) whose potential function is convex. The number of iterations is fixed at 200 in all experiments. When the step size becomes large, the evolution process is accelerated, leading to over-smoothing. The results of AD and RAD show similar behaviors. They diverge when the evolution step size is larger than 0.25, as mentioned in (47). The results of RAD are smoother than that of AD, because the gradient is calculated in an image regularized (smoothed) by a Gaussian filter. AMD shows better performance than AD and RAD, but has artifacts when the evolution step size is 0.25. Furthermore, as the median filter is repeatedly applied, the impulsive noise is eliminated, but the image tends to be over-smoothed. RSF shows the best performance in the sense that the results are well-localized and artifact-free. By considering the fact that the number of iteration and the evolution step size are same, at least RSF can preserve the features, as well as the image structure. As time evolves, the image is flattened, so that other filters act like conventional Gaussian filter, that is, isotropic diffusion. Meanwhile, the velocity of diffusion automatically decreases in RSF when the features begin to be flattened, which leads to preserve the image structure.

Fig. 8 also shows the results for the *Lena* image corrupted by mixture noise in Fig. 6(a). The parameters are the same as those in Fig. 7, except that the number of iterations is 500 and an edge stopping function in (11) is used whose potential function is nonconvex. AD and RSF show similar behaviors in terms of edge-preserving smoothing, except the capability of handling impulsive noise. As expected, Gaussian noise is handled accurately by AD, but impulsive noise still exists even when the number of iteration is large (=500) and an evolution step size increases (0.25–1.0). Since the nonconvex potential function is used, the inverse diffusion occurs in AD when the intensity difference between neighboring pixels is larger than threshold $\lambda$ in (11). Namely, outliers by an impulsive noise as well as edges are assumed to be preserved. The results of AMD have no impulsive noise since it has a discriminant power for handling impulsive noise, that is, median filter. It shows, however, artifact and localization error when evolution step size is set to 0.25 or 0.5 (see black spots in *Lena*'s hat). Furthermore, although an impulsive noise is eliminated at early evolution stage, it over-smooths an image, since median filter

TABLE I

OBJECT EVALUATION FOR THE PROPOSED METHOD

| | | | *Lena* (dB) | *House* (dB) | *Brain* (dB) | Avg. PSNR | Avg. proc. time (ms) | Nor. proc. time |
|---|---|---|---|---|---|---|---|---|
| Noise type A:<br>AWGN with standard deviation, 20 | Input noisy image | $\tau$ | 22.32 | 22.33 | 23.06 | 22.57 | | |
| Method | AD [5] | 0.25 | 29.54 | 31.33 | 28.47 | 29.78 | 3.40 | 0.84 |
| | | 0.50 | 29.53 | 31.24 | 28.46 | 29.74 | | |
| | | 1.00 | 29.22 | 30.59 | 28.31 | 29.37 | | |
| | RAD [8] | 0.25 | 29.51 | 31.51 | 29.14 | 30.05 | 8.00 | 1.96 |
| | | 0.50 | 25.37 | 25.22 | 25.84 | 25.48 | | |
| | | 1.00 | 18.47 | 18.21 | 19.93 | 18.87 | | |
| | AMD [19] | 0.25 | 28.06 | 29.88 | 27.72 | 25.53 | 16.00 | 3.93 |
| | | 0.50 | 28.17 | 30.02 | 27.81 | 28.67 | | |
| | | 1.00 | 28.31 | 30.24 | 27.93 | 28.83 | | |
| | RSF | 0.25 | 28.98 | 30.97 | 28.56 | 29.50 | 4.07 | 1.00 |
| | | 0.50 | 28.97 | 30.96 | 28.57 | 29.50 | | |
| | | 1.00 | 28.72 | 30.73 | 28.47 | 29.31 | | |
| Noise type B:<br>Impulsive noise with density, 0.05 | Input noisy image | $\tau$ | 20.63 | 20.66 | 19.84 | 20.38 | | |
| Method | AD [5] | 0.25 | 25.44 | 27.48 | 23.17 | 25.36 | 3.50 | 0.85 |
| | | 0.50 | 25.41 | 27.32 | 13.16 | 25.30 | | |
| | | 1.00 | 25.26 | 26.88 | 23.17 | 25.10 | | |
| | RAD [8] | 0.25 | 27.86 | 29.82 | 26.79 | 28.16 | 8.01 | 1.95 |
| | | 0.50 | 25.45 | 26.41 | 25.39 | 25.75 | | |
| | | 1.00 | 21.93 | 22.79 | 22.25 | 22.32 | | |
| | AMD [19] | 0.25 | 31.55 | 34.14 | 31.17 | 32.29 | 16.03 | 3.91 |
| | | 0.50 | 31.55 | 34.19 | 31.07 | 32.27 | | |
| | | 1.00 | 31.39 | 34.02 | 30.77 | 32.06 | | |
| | RSF | 0.25 | 30.27 | 31.80 | 28.91 | 30.33 | 4.10 | 1.00 |
| | | 0.50 | 30.66 | 32.11 | 29.49 | 30.75 | | |
| | | 1.00 | 31.23 | 32.82 | 30.60 | 31.55 | | |
| Noise type A+B:<br>AWGN in A and impulsive noise in B | Input noisy image | $\tau$ | 18.41 | 18.43 | 18.15 | 18.33 | | |
| Method | AD [5] | 0.25 | 25.39 | 27.42 | 22.93 | 25.25 | 3.45 | 0.86 |
| | | 0.50 | 25.35 | 27.25 | 22.91 | 25.17 | | |
| | | 1.00 | 25.12 | 26.70 | 22.89 | 24.90 | | |
| | RAD [8] | 0.25 | 27.09 | 29.04 | 25.65 | 27.26 | 8.21 | 2.05 |
| | | 0.50 | 22.94 | 22.89 | 22.04 | 22.62 | | |
| | | 1.00 | 17.73 | 17.62 | 18.41 | 17.92 | | |
| | AMD [19] | 0.25 | 27.85 | 29.65 | 27.39 | 28.30 | 15.89 | 3.97 |
| | | 0.50 | 27.95 | 29.81 | 27.47 | 28.41 | | |
| | | 1.00 | 28.09 | 30.01 | 27.57 | 28.56 | | |
| | RSF | 0.25 | 27.74 | 29.88 | 26.85 | 28.16 | 3.99 | 1.00 |
| | | 0.50 | 27.81 | 29.91 | 27.10 | 28.27 | | |
| | | 1.00 | 27.75 | 29.80 | 27.28 | 28.28 | | |

Avg. PSNR: Average PSNR
Avg. Proc. Time: Average processing time per iteration
Nor. Proc. Time: Normalized processing time

is applied in an iterative manner. The results of RAD have no outliers since Gaussian regularization suppresses the outlier, but are over-smoothed as well. Also, it diverges when large evolution step size ($\tau = 1.0$) is used. In contrast, RSF handles the impulsive noise as well as Gaussian noise accurately for all evolution step sizes, although it does not perform well when outliers are concentrated densely, since it uses a small kernel, as in (38).

*2) Quantitative Comparison:* To compare the performance quantitatively, the PSNR is measured against varying evolution step sizes and the number of iterations, as shown in Fig. 9. The results for the *Lena* image with the mixture noise are obtained with (12) only, since the results obtained with (11) show similar behavior. Fig. 9 demonstrates that the result of RSF is less sensitive to the number of iterations and evolution step size. As expected, the performance of RAD and AMD is rapidly degraded when the evolution step size $\tau = 1.0$ is larger than 0.25. The performance of AD is also inferior to that of RSF for all the evolution step sizes. For instance, when $\tau = 1.0$, the PSNR of AD is averagely 2.5 dB smaller than that of RSF.

Table I shows the PSNR and processing time of AD, RAD, AMD, and RSF with a varying evolution step size when the edge stopping function in (12) is used. Experiments are conducted with various types of noise: AWGN, impulsive noise, and mixture noise. The “avg. PSNR” and “avg. proc. time” denote the average PSNR and the average processing time per single iteration. The “nor. complexity” represents a relative ratio of average processing time when that of the RSF is normalized to 1.0.

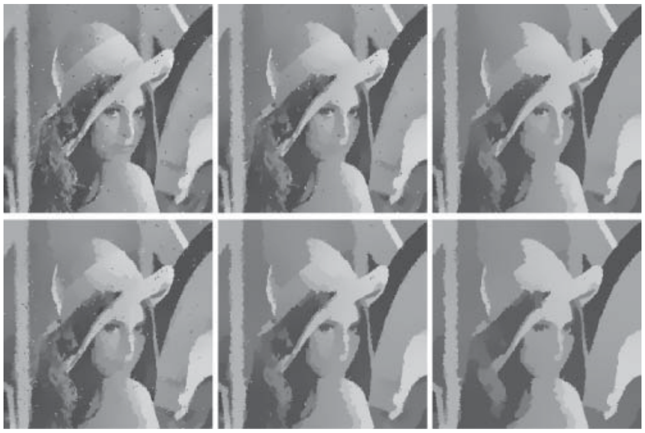


Fig. 10. Results for the *Lena* image with mixture noise in Fig. 6(a). RSF (from top to bottom) with 8-neighborhood and with 24-neighborhood. The parameters are the same as those in Fig. 8. Please see the electronic version for better visibility.

As expected, since robustness is not inherently imposed on AD, AD shows poor results for impulsive noise and mixture noise, while obtaining the best results for AWGN. The performance of RAD is the worst since it diverges when the evolution step size is larger than 0.25. It is worthy to note that when the evolution step size is 0.25, RAD shows better performance than AD for all types of noise since regularization is somewhat effective to outliers. AMD shows the best performance for impulsive noise and mixture noise since it inherently contains a detector and remover for the impulsive noise. It is not suitable, however, to scale-space filtering as shown in Figs. 7 and 8. Namely, it shows a poor performance for AWGN since the median filter, which is not optimal to AWGN, is repeatedly applied. In contrast, the performance of the RSF is relatively less sensitive to the evolution step size and noise type. It shows better performance than AMD in AWGN, and is comparable with AMD in mixture noise. Moreover, considering the complexity, it shows the best performance. The complexity of RSF is comparable with that of AD, because RSF only needs four additions and one division, while including operations for AD per single iteration. RAD and AMD require Gaussian regularization and median filtering per single iteration, making the filtering more complex.

In conclusion, RSF shows consistent performance regardless of both the evolution step size and the noise type, and its complexity is relatively low.

### C. Enlarging the Kernel Size

When the nonconvex function such as (11) is used as an "edge-stopping" function, some impulsive noises still remain even in the results even filtered by the RSF as shown in Fig. 8. This can be further remedied by enlarging the kernel size as in [26]. Fig. 10 shows the results, obtained using (from top to bottom) 8- or 24-neighborhood, for the *Lena* image with the mixture noise in Fig. 6(a). The parameters are the same as those in Fig. 8 except the kernel size. It is worthy of noting that the evolution step size for ensuring the numerical stability of the RSF does not depend on the kernel size, that is, $0 \leq \tau \leq 1$, whereas the evolution step size of the AD should be altered according to the kernel size, that is, $0 \leq \tau \leq 1/N$ when $N$-neighborhood is used (see Section III-C). Comparing with the results of Fig. 8, we could get two observations. First, the RSF with the large kernel is helpful to dealing with the outliers, since more information can be aggregated. Second, the RSF with the large evolution step size is advantageous, since the evolution process is more accelerated. The processing time, however, increases linearly with the kernel size. For instance, the processing time per iteration with 8-neighborhood (24-neighborhood) is nearly twice (five times) of that with 4-neighborhood.

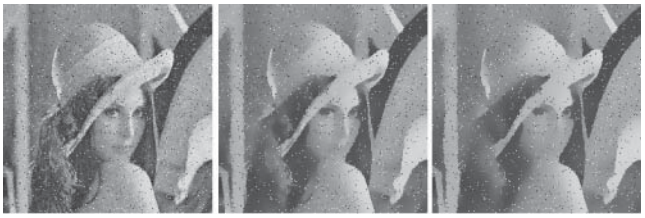


Fig. 11. Behavior of the bilateral filter for the *Lena* image with mixture noise in Fig. 6(a). The number of iterations is set to (from left to right) 1, 5, and 10. The Gaussian kernel with a size 11 × 11 is used. The bandwidth parameter is set to the standard deviation of the AWGN, 20. Please see the electronic version for better visibility.

### D. Comparison With Bilateral Filter

The bilateral filter is a typical example of the nonlinear smoothing that regularizes homogeneous regions and relatively unimportant features while preserving universal features [21], [27].

The RSF is similar to the bilateral filter in that both methods contain the normalization term. The bilateral filter, however, is not robust to various outliers as described in Section IV-D, since it is theoretically equivalent to the single iteration solution of the weighted least square optimization [27]. Fig. 11 shows the behavior of the bilateral filter applied to the noisy image of Fig. 6(a). The bilateral filter is repeatedly applied for speculating the filtering behavior against the outliers. The number of iteration is set to (from left to right) 1, 5, and 10, respectively. Although the bilateral filter can successfully handle the AWGN, it cannot remove the impulsive noise even after multiple iterations.

## VI. Discussion and Conclusion

*1) Summary:* Robust scale-space filter has been proposed from second-order PDEs. It adaptively changes the amount of flux according to the local topology of the neighborhood. Similar to modeling the heat or temperature flows in physics, it was derived by coupling Fick's law with a generalized continuity equation in which a source or sink is modeled via a specific heat capacity.

*2) Filter Design:* Table II shows the family of the scale-space filtering with second-order PDEs. Note that we can exploit the relationship between the robust scale-space filter and existing filters by adaptively changing $c(x, y, t)$ and $n(x, y, t)$. Therefore, it is feasible to design a new filter by altering the function $c(x, y, t)$ or $n(x, y, t)$ in second-order PDEs. Please refer to Appendix, which derives a relationship with the Lee filter [25].

TABLE II
SECOND-ORDER PDE FOR SCALE-SPACE FILTERING

$$n(x,y,t)\partial_t I(x,y,t) - c(x,y,t)\Delta I(x,y,t) - \nabla c(x,y,t)\cdot\nabla I(x,y,t) = 0$$

| $n(x,y,t)$ | $c(x,y,t)$ | Scale-space filtering |
|---|---|---|
| 1 | $c$ (constant) | $\partial_t I(x,y,t) = c\Delta I(x,y,t)$: Isotropic diffusion |
| 1 | $g\left(\lVert\nabla I(x,y,t)\rVert\right)$ | $\partial_t I(x,y,t) = \nabla\cdot\left[g\left(\lVert\nabla I(x,y,t)\rVert\right)\nabla I(x,y,t)\right]$: AD |
| $\lvert\nabla\cdot[c(x,y,t)\nabla A(x,y,t)]\rvert$ | $g\left(\lVert\nabla I(x,y,t)\rVert\right)$ | $\partial_t I(x,y,t) = \frac{\nabla\cdot[g(\lVert\nabla I(x,y,t)\rVert)\nabla I(x,y,t)]}{\lvert\nabla\cdot[g(\lVert\nabla I(x,y,t)\rVert)\nabla A(x,y,t)]\rvert}$: RSF |

*3) Extension:* Robust scale-space filter can be combined with more powerful methods. First, it can be applied to vector image representation for a color image in a similar way to [17], or tensor representation [13]–[17]. It leads to edge or coherence enhancing filtering which is robust to outliers. Second, it can be extended to higher order filtering such as anisotropic fourth-order diffusion that preserves ramp and step edges simultaneously [29]. Third, the robust scale-space filter can be combined with existing filters, which makes them more powerful. One feasible example is a regularized robust scale-space filter which combines the proposed method with RAD [8]. It can also be combined with other AD methods for handling speckle noise [18]. Also, the robust scale-space filter does not have a discriminant power to detect outliers explicitly. It can be addressed by combining it with other outlier handling methods [10], [23], [24].

*4) Acceleration:* The robust scale-space filter can be accelerated by using a semi-implicit or an implicit schemes without losing the numerical stability. The (semi-) implicit scheme has been known to be more complex yet stable than the explicit scheme. This enables the filter to have a larger step size, making it converge with a smaller number of iteration [30], [31]. Furthermore, it is similar to the bilateral filter [21] containing the normalization term, giving us the chance to accelerate the proposed method via signal processing techniques widely used for the fast bilateral filtering [32]. Namely, we can utilize the (semi-) implicit scheme for reducing the number of iteration and the signal processing techniques for reducing the computational complexity per iteration.

*5) Parameter Optimization:* It was shown that the robust scale-space filter is superior to the conventional diffusion in the same environments. The parameters used in the experiment, however, were not optimized. It is expected that the performance of the robust scale-space filter could be improved with optimized parameters [33].

## APPENDIX
## RELATIONSHIP WITH THE LEE FILTER [18]

The Lee filter is one of the most representative techniques for eliminating speckle noise in ultrasonic or radar images, while preserving features [25]. Although the original Lee filter is a one-step method, it can be represented in iterative fashion as follows:

$$I_{i,j}^{t+1} = \bar{I}_{i,j}^{t} + k_{i,j}^{t}\left(I_{i,j}^{t} - \bar{I}_{i,j}^{t}\right) \tag{61}$$

where $\bar{I}_{i,j}^{t}$ denotes a mean value within the neighborhood. The filter coefficient is represented by $k_{i,j}^{t}$, which adapts with respect to the local statistics

$$k_{i,j}^{t} = 1 - \frac{(C_u)^2}{\left(C_{i,j}^{t}\right)^2} \tag{62}$$

where

$$\left(C_{i,j}^{t}\right)^2 = \frac{(1/|N|)\sum_{(k,l)\in N(i,j)}\left(I_{k,l}^{t} - \bar{I}_{i,j}^{t}\right)^2}{\left(\bar{I}_{i,j}^{t}\right)^2}. \tag{63}$$

$|N|$ denotes the number of the neighborhood. $(C_u)^2$ is a constant, which can be defined by using the effective number of looks $(ENL)$ of the noisy image, or the variance $(\mathrm{var}_h)$ and mean $(\bar{m}_h)$ value of homogeneous regions as follows:

$$(C_u)^2 = \frac{1}{ENL} \quad \text{or} \quad \frac{\mathrm{var}_h}{\bar{m}_h}. \tag{64}$$

Let us assume that the neighborhood of the Lee filter is equal to that of the robust scale-space filter. Equation (61) can then be expressed as follows:

$$\begin{aligned} I_{i,j}^{t+1} &= I_{i,j}^{t} + \left(1 - k_{i,j}^{t}\right)\left(\bar{I}_{i,j}^{t} - I_{i,j}^{t}\right) \\ &= I_{i,j}^{t} + \frac{\left(1 - k_{i,j}^{t}\right)}{|N(i,j)|}\sum_{(k,l)\in N(i,j)}\left(I_{k,l}^{t} - I_{i,j}^{t}\right). \end{aligned} \tag{65}$$

Yu and Acton showed that there is a discrete implementation of the isotropic diffusion [18] when the filter coefficient $k_{i,j}^{t}$ (or $C_{i,j}^{t}$) is a constant. In contrast to isotropic diffusion that smooths an image evenly without considering the local structure, the Lee filter can change the filter coefficients adaptively according to the local characteristics.

Plugging (63) to (62) leads to

$$1 - k_{i,j}^{t} = (C_u)^2\,|N(i,j)|\left[\frac{\left(\bar{I}_{i,j}^{t}\right)^2}{\sum_{(k,l)\in N(i,j)}\left(I_{k,l}^{t} - \bar{I}_{i,j}^{t}\right)^2}\right]. \tag{66}$$

Equation (65) can then be represented as follows:

$$I_{i,j}^{t+1} = I_{i,j}^{t} + (C_u)^2\left(\bar{I}_{i,j}^{t}\right)^2\left[\frac{\sum_{(k,l)\in N(i,j)}\left(I_{k,l}^{t} - I_{i,j}^{t}\right)}{\sum_{(k,l)\in N(i,j)}\left(I_{k,l}^{t} - \bar{I}_{i,j}^{t}\right)^2}\right]. \tag{67}$$

If we assume that the intensity within the neighborhood $N(i,j)$ changes smoothly, (67) can be approximated as

$$I_{i,j}^{t+1} \approx I_{i,j}^{t} + (C_u)^2 \left(\bar{I}_{i,j}^{t}\right)^2 \left[ \frac{\sum_{(k,l)\in N(i,j)} \left(I_{k,l}^{t} - I_{i,j}^{t}\right)}{\sum_{(k,l)\in N(i,j)} \left(I_{k,l}^{t} - I_{i,j}^{t}\right)^2} \right]. \tag{68}$$

It is a discrete implementation of (1) with

$$n(x,y,t) = \nabla \cdot [d(x,y,t)\nabla A(x,y,t)] \tag{69}$$

$$c(x,y,t) = 1 \tag{70}$$

where

$$d(x,y,t) = z\left(\|\nabla I(x,y,t)\|\right) \tag{71}$$

$$z(x) = x^2. \tag{72}$$

That is, (68) is an approximation of the following equation:

$$\partial_t I(x,y,t) = \frac{\Delta I(x,y,t)}{\nabla \cdot [d(x,y,t)\nabla A(x,y,t)]} \tag{73}$$

with an evolution step size as in (74)

$$\tau = (C_u)^2 \left(\bar{I}_{i,j}^{t}\right)^2. \tag{74}$$

## Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

## References

[1] A. P. Witkin, "Scale space filtering," in *Proc. Int. Joint Artif. Intell. Conf.*, 1983, pp. 1019–1022.

[2] J. J. Koenderink, "The structure of images," *Biol. Cybernet.*, vol. 50, no. 5, pp. 363–370, 1984.

[3] R. Hummel and R. Moniot, "Reconstructions from zero crossings in scale space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 12, pp. 245–295, Dec. 1989.

[4] J. Clark, "Singularity theory and phantom edges in scale space," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 5, pp. 720–727, Sep. 1988.

[5] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[6] Y. L. You, W. Xu, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Trans. Image Process.*, vol. 5, no. 11, pp. 1539–1553, Nov. 1996.

[7] J. Weickert, "A review of nonlinear diffusion filtering," in *Proc. Scale Space Theory Comput. Vis.*, 1997, pp. 1–28.

[8] F. Catte, P. L. Lions, J. M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, vol. 29, no. 1, pp. 182–193, 1992.

[9] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 421–432, Mar. 1998.

[10] P. Charbonnier, L. B. Feraud, C. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 298–311, Feb. 1997.

[11] P. Mrazek, J. Weickert, and A. Bruhn, "On robust estimation and smoothing with spatial and tonal kernels," in *Geometric Properties for Incomplete Data*. New York: Springer-Verlag, 2006, pp. 335–352.

[12] R. A. Carmona and S. Zhong, "Adaptive smoothing respecting feature directions," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 353–358, Mar. 1998.

[13] S. D. Zenzo, "A note on the gradient of a multi-image," *Comput. Vis., Graph., Image Process.*, vol. 33, no. 1, pp. 116–125, 1986.

[14] J. Weickert, "Theoretical foundations of anisotropic diffusion in image processing," in *Theoretical Foundations of Computer Vision*, vol. 11, W. Kropatsch, R. Klette, and F. Solina, Eds. New York: Springer-Verlag, 1996, pp. 221–236.

[15] J. Weickert, "Coherence-enhancing diffusion filtering," *Int. J. Comput. Vis.*, vol. 31, nos. 2–3, pp. 111–127, 1999.

[16] M. Felsberg, "Autocorrelation-driven diffusion filtering," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1797–1806, Jul. 2011.

[17] D. Tschumperle and R. Deriche, "Vector-valued image regularization with PDEs: A common framework for different applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 506–517, Apr. 2005.

[18] Y. You and S. T. Acton, "Speckle reducing anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 11, no. 11, pp. 1260–1270, Nov. 2002.

[19] J. Ling and A. C. Bovik, "Smoothing low-SNR molecular images via anisotropic median-diffusion," *IEEE Trans. Med. Imag.*, vol. 21, no. 4, pp. 377–384, Apr. 2002.

[20] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Apr. 2001, pp. 355–363.

[21] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int. Conf. Comput. Vis.*, 1998, pp. 839–846.

[22] E. Kreyszig, *Advanced Engineering Mathematics*, 9th ed. New York: Wiley, 2006, pp. 552–560.

[23] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1747–1754, Nov. 2005.

[24] G. Pok, J. Liu, and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Trans. Image Process.*, vol. 12, no. 1, pp. 85–92, Jan. 2003.

[25] J.-S. Lee, "Digital image enhancement and noise filtering by use of local statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 2, no. 2, pp. 165–168, Mar. 1980.

[26] D. Barash and D. Comaniciu, "A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift," *Image Vis. Comput.*, vol. 22, no. 1, pp. 73–81, 2004.

[27] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1141–1151, Oct. 2002.

[28] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2005.

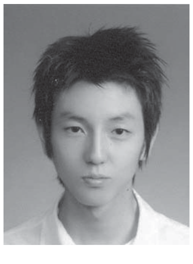
[29] M. R. Hajiaboli, "An anisotropic fourth-order diffusion filter for image noise removal," *Int. J. Comput. Vis.*, vol. 92, no. 2, pp. 177–191, 2010.

[30] J. Weickert, B. M. H. Romeny, and M. A. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 398–410, Mar. 1998.

[31] J. M. Duarte-Carvajalino, P. E. Castillo, and M. Velez-Reyes, "Comparative study of semi-implicit schemes for nonlinear diffusion in hyperspectral imagery," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1303–1314, May 2007.

[32] M. M. Bronstein, "Lazy sliding window implementation of the bilateral filter on parallel architectures," *IEEE Trans. Image Process.*, vol. 20, no. 6, pp. 1751–1756, Jun. 2011.

[33] P. Mrazek and M. Navara, "Selection of optimal stopping time for nonlinear diffusion filtering," *Int. J. Comput. Vis.*, vol. 52, nos. 2–3, pp. 189–203, 2003.



**Bumsub Ham** (S'09) received the B.S. degree in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2008, where he is currently pursuing the joint M.S. and Ph.D. degrees.

His current research interests include variational methods and PDEs to solve 3-D computer vision and image processing problems, particularly stereo vision, 3-D modeling, super-resolution, and HDR imaging.

**Dongbo Min** (M'09) received the B.S., M.S., and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2003, 2005, and 2009, respectively.

He was with the Mitsubishi Electric Research Laboratories, Cambridge, MA, as a Post-Doctoral Researcher, from June 2009 to June 2010. He is currently with the Advanced Digital Sciences Center, Singapore, which was jointly founded by the University of Illinois at Urbana-Champaign, Urbana, and the Agency for Science, Technology, and Research, a Singapore government agency. His current research interests include 3-D computer vision, computational photography, 3-D modeling, and hybrid sensor systems.

**Kwanghoon Sohn** (M'92) received the B.E. degree in electronic engineering from Yonsei University, Seoul, Korea, in 1983, the M.S.E.E. degree in electrical engineering from the University of Minnesota, Minneapolis, in 1985, and the Ph.D. degree in electrical and computer engineering from North Carolina State University, Raleigh, in 1992.

He was a Senior Member of the research staff with the Satellite Communication Division, Electronics and Telecommunications Research Institute, Daejeon, Korea, from 1992 to 1993, and a Post-Doctoral Fellow with the MRI Center, Medical School of Georgetown University, Washington, DC, in 1994. He was a Visiting Professor with Nanyang Technological University, Singapore, from 2002 to 2003. He is currently a Professor with the School of Electrical and Electronic Engineering, Yonsei University. His current research interests include 3-D image processing, computer vision, and image communication.

Dr. Sohn is a member of the Society of Photo-Optical Instrumentation Engineers.