

NON-LOCAL IMAGE SMOOTHING BY APPLYING ANISOTROPIC DIFFUSION PDE'S IN THE SPACE OF PATCHES

D. Tschumperlé L. Brun

GREYC (CNRS UMR 6072), 6 Bd du Maréchal Juin, 14050 Caen Cedex, France.

ABSTRACT

We design a family of non-local image smoothing algorithms which approximate the application of diffusion PDE's on a specific Euclidean space of image patches. We first map a noisy image onto this high-dimensional space and estimate its geometric structure thanks to a straightforward extension of the structure tensor field. The tensors spectral elements allows us to design an oriented high-dimensional smoothing process by the means of anisotropic regularization PDE's which have both local and non-local properties and whose solutions are estimated by locally oriented high-dimensional convolutions. We show that the *Bilateral Filtering* and *Non-Local Means* methods are the isotropic cases of our denoising framework.

Index Terms— Edge-preserving denoising, non-local smoothing, anisotropic diffusion, regularization PDE's.

1. INTRODUCTION

Over recent years, there has been a high raise of interest for methods based on patch analysis in the image processing and computer vision communities. A patch, basically defined as a (usually square) local neighborhood of an image region, is a very basic model to work with. But, it has the intrinsic ability to catch large-scale structures in images. For instance, using patches provides one simple way to compare texture informations at different locations, as soon as the patch sizes are higher than the maximal texel size (smallest significant unit of a texture). Moreover, patch-based algorithms are somehow intuitive : They mostly reproduce the way humans would manage image processing tasks by analyzing, comparing and manipulating semi-local image neighborhoods together. Surprisingly, these algorithms are often simple to implement but generally perform very well. Patch-based methods have been known to give interesting solutions when it comes to estimate the displacement field between two images (*Block Matching* algorithm) [10], synthesize textures from input models [1, 9, 22], transfer textures from one image to another one [2, 12] or perform texture-preserving inpainting [6, 11].

Impressive results have been recently obtained by patch-based methods in the field of image denoising, originated by the *Non Local Means* framework [5] and continued with various derivatives [4, 7, 13]. Mostly, these methods proceed by iteratively computing a weighted average of patches at different locations. The outstanding quality of the denoising results outputted by such algorithms made them rapidly enter the hall of fame of image smoothing techniques.

Clearly, patch-based denoising methods have become very serious competitors to other well established regularization techniques, such as the ones based on diffusion PDE's. One of the major advantage of the formers are their intrinsic non-locality : they are able to integrate large spatial informations and catch local image areas with auto-similarities. As noise is a random stochastic process, having several samples of similar image structures definitely helps in

reconstructing artefacts-free versions of degraded images [13]. On the contrary, PDE regularization methods act on a more local scale since they mimic physical processes by locally diffusing pixel values along the image structures [15, 18, 20, 23]. But this locality goes along a very precise control on how the smoothing is achieved. The anisotropic behavior of these PDE's is perfectly adjustable and is able to fit specific constraints related to any targeted application. This certainly explains part of the success they met until this time.

Combining the best of these two worlds is a very exciting and challenging purpose. We propose to move toward this direction by considering an alternate *high-dimensional patch space* on which the degraded image is mapped. In this space, each existing image patch is associated to a single point. The mapping is done continuously such that the geometry of the obtained patch surface reflects both the local and non-local structural informations of the image data (section 2). The geometric features of this patch-projected dataset are then retrieved by computing a high-dimensional extension of the structure tensor field [8, 23] (section 3). This naturally leads to the design of a diffusion tensor-directed smoothing process, expressed as the evolution of a high-dimensional multi-valued anisotropic diffusion PDE on the Euclidean patch space.

From a practical point of view, we show that this evolution can be approximated by several local convolutions that can be computed in the initial image domain, so that storing the entire patch space is never required. We also exhibit the particular isotropic cases of our framework which are found to be in fact variants of the well known *Non-Local Means* and *Bilateral Filtering* schemes [5, 19] (section 4). Some application results and comparisons for the problem of color image denoising conclude this paper (section 5).

2. DEFINITION OF THE PATCH SPACE

Let us consider a 2D noisy image $\mathbf{I} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^n$ (here $n = 3$, for color images). The i^{th} component of a vector or image \mathbf{X} will be denoted as X_i . We define a patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ as the set of all image values belonging to a spatially discretized local $p \times p$ neighborhood of \mathbf{I} , centered at (x, y) . For simplification sakes, the spatial discretization step of a patch is assumed to be 1. The dimension p is considered as odd, *i.e.* $p = 2q + 1$ ($q \in \mathbb{N}$). Actually, a patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ can be ordered as a np^2 -dimensional vector as :

$$\mathcal{P}_{(x,y)}^{\mathbf{I}} = (I_{1(x-q,y-q)}, \dots, I_{1(x+q,y+q)}, I_{2(x-q,y-q)}, \dots, I_{n(x+q,y+q)})$$

A multi-valued patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ is thus the concatenation of the scalar patches $\mathcal{P}_{(x,y)}^{I_i}$ for all image channels $i = 1..n$. The mathematical study of the whole manifold geometry formed by these image patches has been initiated in [16]. Here we will rather consider this patch-based geometry of the image at a more local scale.

3. PATCH-BASED GEOMETRY IN Γ

Mapping in the patch space : We define the $(np^2 + 2)$ -dimensional patch space $\Gamma = \Omega \times \mathbb{R}^{np^2}$. Each point \mathbf{p} of Γ is a high-dimensional vector which contains both informations of any (x, y) coordinates in Ω and of all values of any $p \times p$ patch \mathcal{P} in \mathbb{R}^{np^2} . Obviously, we want to highlight the points $\mathbf{p} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$ in Γ , *i.e.* the locations which precisely correspond to existing patches in \mathbf{I} (*a.k.a.* the *located patches* of \mathbf{I}). We define the function $\tilde{\mathbf{I}}$ in Γ such that $\tilde{\mathbf{I}}_{(\mathbf{p})}$ is non zero only for these located patches : $\tilde{\mathbf{I}} : \Gamma \rightarrow \mathbb{R}^{np^2+1}$, s.a. $\forall \mathbf{p} \in \Gamma$,

$$\tilde{\mathbf{I}}_{(\mathbf{p})} = \begin{cases} (\mathbf{I}_{(x,y)}, 1) & \text{if } \mathbf{p} = (x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}}) \\ 0 & \text{elsewhere} \end{cases} \quad (1)$$

The application \mathcal{F} such that $\tilde{\mathbf{I}} = \mathcal{F}(\mathbf{I})$ computes a patch-based representation of \mathbf{I} . Note that the value space of $\tilde{\mathbf{I}}$ has an extra component set to 1 at the located patches of \mathbf{I} . This is comparable to what is usually done when dealing with projective spaces : This value plays a role of weighting action when inverting \mathcal{F} , *i.e.* retrieving back \mathbf{I} from $\tilde{\mathbf{I}}$. Intuitively, it defines how much a patch in Γ is *meaningful*, and by default all located patches of the initial image have the same importance. Note that $\tilde{\mathbf{I}}$ is a discontinuous multi-valued function. To avoid further derivation problems, we will in fact consider $\tilde{\mathbf{I}}_\epsilon = \mathbf{I} * G_\epsilon$, a continuous version of $\tilde{\mathbf{I}}$, where each patch $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ of \mathbf{I} is mapped as a normalized Gaussian function G_ϵ with a variance ϵ close to 0, instead of a single Dirac point in Γ .

Back-projection on the image domain : Due to the high dimensionality of Γ , there are of course no unique ways to invert the patch transform $\tilde{\mathbf{I}} = \mathcal{F}(\mathbf{I})$. We define a back-projection method based on two steps : First, we retrieve the most significant patch $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ in Γ for every location (x, y) in Ω .

This is the one with the maximum projective weight, *i.e.*

$$\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}} = \operatorname{argmax}_{\mathbf{q} \in \mathbb{R}^{np^2}} \tilde{I}_{np^2+1}(x, y, \mathbf{q}) \quad (2)$$

Note that if one perturbs only slightly the patch representation $\tilde{\mathbf{I}}$ of \mathbf{I} , one will likely find that the significant patches $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ stays at the same locations $\mathcal{P}_{(x,y)}^{\mathbf{I}}$ as the original ones, even though the pixel values of these most significant patches may have been modified.

In a second step, the back-projected image $\hat{\mathbf{I}}$ is reconstructed by combining the most significant patches together. Here, we use the simplest possible strategy, *i.e.* copy the normalized center pixel of each $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ at its corresponding location (x, y) :

$$\forall (x, y) \in \Omega, \hat{I}_{(x,y)} = \frac{\tilde{I}_{ip^2 + \frac{p^2+1}{2}}(x, y, \mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}})}{\tilde{I}_{np^2+1}(x, y, \mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}})} \quad (3)$$

We could also have copied an entire sub-patch of $\mathcal{P}_{sig(x,y)}^{\tilde{\mathbf{I}}}$ while overlapping neighborhood patches according to their relative weights. These patch copying/averaging considerations frequently appear in the patch-based algorithm literature. The best pick probably depends on the targeted application.

This simple patch transform allows to project at the same time the local and non-local structural informations of an image \mathbf{I} onto an Euclidean space Γ where nearby and non-zero points correspond to patches that are similar in values (with respect to the L_2 norm) or are spatially grouped. Therefore, applying a local process (by nature) on Γ followed by a back-projection procedure is a quite elegant way of embedding it with non-local properties. This is what we aim to do in the followings, with the proposal of an edge-preserving filtering based on the evolution of diffusion PDE's on Γ .

Smoothing $\tilde{\mathbf{I}}$ anisotropically requires a preliminary analysis of its local geometry in Γ , by finding the directions and the contrast of the discontinuities in $\tilde{\mathbf{I}}$. For classical multi-valued images, this step is usually done by retrieving the spectral elements of the (possibly smoothed) structure tensor field $\tilde{\mathbf{J}}_\sigma$, as suggested in [8, 23]. But, as the expression of $\tilde{\mathbf{J}}_\sigma$ is dimension-independent, we can easily consider its high-dimensional extension $\tilde{\mathbf{J}}_\sigma = \sum_{i=1}^n \nabla \tilde{I}_{i\sigma} \nabla \tilde{I}_{i\sigma}^T$ for our patch transform dataset $\tilde{\mathbf{I}}$, where $\tilde{I}_{i\sigma} = \tilde{I}_i * G_\sigma$ is a filtered version of \tilde{I}_i by a normalized $(np^2 + 2)$ -dimensional Gaussian kernel G_σ .

Estimation of $\nabla \tilde{\mathbf{I}}_\sigma$: Practically, the high dimensionality $(np^2 + 2)$ of Γ is a major obstacle for the estimation of $\tilde{\mathbf{J}}_\sigma$ since it is not possible to store all values of $\tilde{\mathbf{I}}$ in computer memory. Fortunately, these gradients can be estimated directly on the original image domain Ω thanks to the derivation property of the convolution $\partial(\tilde{\mathbf{I}} * \mathcal{K}) = \tilde{\mathbf{I}} * \partial\mathcal{K}$, and to the fact that $\tilde{\mathbf{I}}$ vanishes almost everywhere except at $(x, y, \mathcal{P}_{(x,y)}^{\mathbf{I}})$. So, we get :

$$\forall \mathbf{p} \in \Gamma, \nabla \tilde{I}_{i\sigma}(\mathbf{p}) = \int_{(k,l) \in \Omega} I_{i(k,l)} \nabla G_{\sigma(\mathbf{p}-\mathbf{q}(k,l))} dk dl$$

where $\mathbf{q}(k,l) = (k, l, \mathcal{P}_{(k,l)}^{\mathbf{I}})$. Once the smoothed gradient $\nabla \tilde{I}_{i\sigma}$ is estimated, the computation of the symmetric $(np^2 + 2) \times (np^2 + 2)$ structure tensor $\tilde{\mathbf{J}}_\sigma$ becomes a straightforward operation.

Geometry analysis : As demonstrated in [8], the spectral elements of the structure tensor $\tilde{\mathbf{J}}_\sigma$ are closely related to the local variations of the studied multi-valued function. In our case, $\mathcal{N}_{(\mathbf{p})} = \sqrt{\operatorname{trace}(\tilde{\mathbf{J}}_{(\mathbf{p})})}$ gives precious informations about the local variation amplitudes considering at the same time both the spatial and patch-valued distributions of the neighboring points (*i.e.* the patch discontinuities). A high value for $\mathcal{N}_{(\mathbf{p})}$ means that the pixel is located on an edge of \mathbf{I} , but also that the patch to which it belongs is probably dissimilar to the other patches within its neighborhood.

The principal eigenvector $\tilde{\mathbf{u}}$ of $\tilde{\mathbf{J}}_\sigma$ is another valuable feature : it gives the orientation normal to the isophotes of $\tilde{\mathbf{I}}$, and here, is closely related to the normal vector to the manifold formed by the located patches of \mathbf{I} in Γ . Clearly, smoothing $\tilde{\mathbf{I}}$ along $\tilde{\mathbf{u}}$ should be avoided as much as possible in order to preserve the significant discontinuities and patch structures of \mathbf{I} . This is the usual edge-preserving heuristic which consists in favoring smoothing along the edge directions instead of across them. Here, the ‘‘edges’’ are simply considered as patch-based features. The variance σ used for the gaussian smoothing of $\tilde{\mathbf{J}}_\sigma$ tells about the regularity of the estimated spatial-patch geometry of $\tilde{\mathbf{I}}$. It ideally depends on the level of noise in \mathbf{I} . Note that the importance of the patch component versus the spatial part is directly proportional to the value range of the image values : In this setting, multiplying the values of \mathbf{I} with a factor $\lambda > 0$ allows to give more importance to the spatial part ($\lambda \rightarrow 0$) or to the patch-based part ($\lambda \gg 0$) of the analyzed image geometry.

4. PATCH-BASED DIFFUSION PDE'S

Now, we have a continuous patch-based representation $\tilde{\mathbf{I}}$ of \mathbf{I} as well as local geometric indicators $\mathcal{N}_{(\mathbf{p})}$ and $\tilde{\mathbf{u}}_{(\mathbf{p})}$. We propose then to apply the following generic diffusion PDE on Γ :

$$\forall \mathbf{p} \in \Gamma, \frac{\partial \tilde{\mathbf{I}}_{(\mathbf{p})}}{\partial t} = \operatorname{trace} \left(\tilde{\mathbf{D}}_{(\mathbf{p})} \tilde{\mathbf{H}}_{i(\mathbf{p})} \right) \quad (4)$$

This is a high-dimensional version of the equation proposed in [20] in the context of classical 2D image regularization. $\tilde{\mathbf{H}}_{i(\mathbf{p})}$ is the

Hessian matrix of \tilde{I}_i at \mathbf{p} . $\tilde{\mathbf{D}} : \Gamma \rightarrow \mathbb{P}_{(np^2+2)}$ is a field of diffusion tensors which direct the regularization process by defining the orientations (tensors eigenvectors) and amplitudes (tensors eigenvalues) of the pixel diffusion at each point \mathbf{p} in Γ . The way $\tilde{\mathbf{D}}$ is chosen is discussed later, but it must obviously depend on $\mathcal{N}_{(\mathbf{p})}$ and $\tilde{\mathbf{u}}_{(\mathbf{p})}$.

Approximation by local convolutions : [20] shows that for a constant tensor field, the PDE (4) is a tensor-deformed heat flow whose solution $\tilde{\mathbf{I}}^{[t]}$ at time t is found to be the convolution of the initial data by a tensor-oriented Gaussian kernel. This result also holds in our patch space : $\tilde{\mathbf{I}}^{[t]} = \tilde{\mathbf{I}}^{[t=0]} * G_{dt}^{\tilde{\mathbf{D}}}$, where $\forall \mathbf{p} \in \Gamma$, $G_{dt}^{\tilde{\mathbf{D}}} = \frac{1}{(4\pi t)^{\frac{np^2+2}{2}}} e^{-\frac{\mathbf{p}^T \tilde{\mathbf{D}}^{-1} \mathbf{p}}{4t}}$.

When $\tilde{\mathbf{D}}$ is *not* constant (usual case), (4) becomes *nonlinear* and getting one explicit solution is no longer possible. To avoid the traditional resolution of (4) using a finite difference scheme, [20] proposed an alternative iterative approximation based on local convolutions by differently oriented Gaussian kernels. In our case, this approximation scheme can be transposed in Γ as :

$$\forall \mathbf{p} \in \Gamma, \quad \tilde{\mathbf{I}}_{(\mathbf{p})}^{[dt]} \approx \int_{\mathbf{q} \in \Gamma} \tilde{\mathbf{I}}_{(\mathbf{q})}^{[t=0]} G_{dt(\mathbf{p}-\mathbf{q})}^{\tilde{\mathbf{D}}(\mathbf{p})} d\mathbf{q} \quad (5)$$

where dt is the discretized time step of the PDE evolution. Note that as (5) is a local averaging process, the approximation remains stable even for large dt . The approximation is also better when $\tilde{\mathbf{D}}$ does not depend on the time t (*i.e.* is estimated from $\tilde{\mathbf{I}}^{[t=0]}$) and is not locally varying too much (it has smoothness properties). It is finally worth noting that (5) will not modify the locations of the most significant patches in $\tilde{\mathbf{I}}$. As a consequence, the back-projection procedure (3) is simplified since $\mathcal{P}_{siq(x,y)}^{\tilde{\mathbf{I}}}$ and $\mathcal{P}_{(x,y)}^{\tilde{\mathbf{I}}}$ stay in place. All this means that the solution of (4) can be actually estimated using a single (large) iteration, only at the original patches locations $(x, y, \mathcal{P}_{(x,y)}^{\tilde{\mathbf{I}}})$, and as $\tilde{\mathbf{I}}^{[t=0]}$ is sparse, this can be directly expressed in the original image domain $\Omega : \forall (x, y) \in \Omega$,

$$\tilde{\mathbf{I}}_{(\mathbf{p}(x,y))}^{[t]} \approx \int_{(k,l) \in \Omega} \mathbf{I}_{(k,l)}^{[t=0]} G_{dt(\mathbf{p}(x,y)-\mathbf{q}(k,l))}^{\tilde{\mathbf{D}}(\mathbf{p}(x,y))} d_k d_l \quad (6)$$

where $\mathbf{p}(x,y) = (x, y, \mathcal{P}_{(x,y)}^{\tilde{\mathbf{I}}})$ and $\mathbf{q}(k,l) = (k, l, \mathcal{P}_{(k,l)}^{\tilde{\mathbf{I}}})$.

The isotropic case : The isotropic case of (4) corresponds to the (np^2+2) -dimensional heat-flow, *i.e.* when the diffusion tensor field is chosen to be everywhere $\tilde{\mathbf{D}}_{(\mathbf{p})} = \mathbf{I}_d$ (identity matrix). In this setting, $G_{dt}^{\tilde{\mathbf{D}}}$ is an isotropic Gaussian kernel. Estimating the heat flow solution in Γ with (6) and back-projecting it on the original image domain Ω with (3) gives then :

$$\forall (x, y) \in \Omega, \quad \hat{\mathbf{I}}_{(\mathbf{p})}^{[t]} = \frac{\int_{(k,l) \in \Omega} \mathbf{I}_{(k,l)}^{[t=0]} w_{(x,y,k,l)} d_k d_l}{\int_{(k,l) \in \Omega} w_{(x,y,k,l)} d_k d_l} \quad (7)$$

where $w_{(x,y,k,l)} = e^{-\frac{(x-k)^2+(y-l)^2}{4t}} e^{-\frac{\|\mathcal{P}_{(x,y)}^{\tilde{\mathbf{I}}} - \mathcal{P}_{(k,l)}^{\tilde{\mathbf{I}}}\|^2}{4t}}$. Here, we find the same expression as the *Non-Local Means* algorithm [5] except that our averaging weights $w_{(x,y,k,l)}$ also take care of the spatial distance between all compared pixels. In addition, if we set $q = 0$ (*i.e.* considering that patches have a size of 1×1), then the weights simply reduces to $w_{(x,y,k,l)} = e^{-\frac{(x-k)^2+(y-l)^2}{4t}} e^{-\frac{\|\mathbf{I}_{(x,y)} - \mathbf{I}_{(k,l)}\|^2}{4t}}$. This defines the natural multi-valued extension (*e.g.* for color images) of the well known *Bilateral Filtering* technique, initially proposed in [19] for the smoothing of gray-valued images.

The anisotropic case : Actually, isotropic diffusion tensors do not take care of the local geometry of the patch-projected dataset $\tilde{\mathbf{I}}$. The diffusion is done in all patch-space orientations, including $\tilde{\mathbf{u}}_{(\mathbf{p})}$, the principal eigenvector of $\tilde{\mathbf{J}}_{\sigma(\mathbf{p})}$. In order to smooth the image while preserving the local spatial and patch structures of $\tilde{\mathbf{I}}$, we rather propose to consider the following field of anisotropic diffusion tensors :

$$\forall \mathbf{p} \in \Gamma, \quad \tilde{\mathbf{D}}_{(\mathbf{p})} = \frac{1}{\mathcal{N}_{(\mathbf{p})}} (\mathbf{I}_d - \tilde{\mathbf{u}}_{(\mathbf{p})} \tilde{\mathbf{u}}_{(\mathbf{p})}^T) \quad (8)$$

Our choice is inspired by the classical *Total Variation* framework for scalar images [17], corresponding to the anisotropic diffusion $\frac{\partial I}{\partial t} = \text{div}(\nabla I / \|\nabla I\|)$ which can be also expressed as $\frac{\partial I}{\partial t} = \text{trace}(\mathbf{D}\mathbf{H})$ with $\mathbf{D} = \frac{1}{\|\nabla I\|} (\mathbf{I}_d - \eta\eta^T)$ and $\eta = \nabla I / \|\nabla I\|$. Intuitively, this choice of diffusion tensors (8) reduce the amplitude of the patch-based diffusion process on locations having high patch-gradients and smoothes the patch structures always along the hyperplane locally tangent to the discontinuities in $\tilde{\mathbf{I}}$. Consequently, the anisotropic diffusion (4) is a *non-local patch-preserving smoothing procedure*. Note that for 1×1 patches ($q = 0$), our method also defines an anisotropic variant of the *Bilateral Filtering* method. Note that actually, any kind of anisotropic diffusion tensor field \mathbf{D} can be considered in our patch-based smoothing framework (4). The best choice is ideally application-dependent and \mathbf{D} may be designed from any other image features or external datasets if necessary.

5. RESULTS AND CONCLUSIONS

We applied our non-local anisotropic diffusion PDE (4) with the tensor field (8), to the problem of color image denoising. The usual (R, G, B) color basis has been considered here. Fig.1 and 2 show how taking care of the image geometry on the patch space Γ allows in turn to better preserve the structures in the original image domain Ω , compared to the corresponding isotropic version of the patch-based filtering process (*i.e.* the *Non-local means* variant and the color *Bilateral Filtering*). More generally, the framework we proposed is potentially able to add non-locality to many image processing techniques, applying them on a patch representation $\tilde{\mathbf{I}}$ of an image \mathbf{I} before doing a back-projection step. It is worth noting that introducing this non-locality does not explicitly require semi-local spatial integrations (as needed in [14]). Of course, this idea is *not limited* to image smoothing or diffusion PDE's, and therefore opens an interesting and original way to unify local and non-local (patch-based) processing into a common formalism.

6. REFERENCES

- [1] Ashikhmin, M.: Synthesizing Natural Textures. Symposium on Interactive 3D Graphics. p.217–226, 2001.
- [2] Ashikhmin, M.: Fast Texture Transfer. IEEE Computer Graphics and Applications, Vol.23 (4), p.38–43, 2003.
- [3] Barash, D.: A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. IEEE PAMI, Vol.24 (6), p.844–847, 2002.
- [4] Brox, T., Cremers, D.: Iterated Non-local Means for Texture Restoration. SSVM. Vol.4485 of LNCS, Springer, 2007.
- [5] Buades, A., Coll, B., Morel, J.M.: A Non-Local Algorithm for Image Denoising. IEEE CVPR, p.60–65, 2005.
- [6] Criminisi, A., Pérez, P., Toyama, K.: Region Filling and Object Removal by Exemplar-based Image Inpainting. IEEE Trans. on Image Proc., Vol.13 (9).



(a) Original grayscale image with texture noise.



(b) Denoised version using our non-local anisotropic diffusion Eq.(4).

Fig. 1. Result of our patch-based anisotropic denoising.

- [7] K. Dabov, A. Foi, and K. Egiazarian : Image restoration by sparse 3D transform-domain collaborative filtering. Proc. SPIE Electr. Imag. '08, no. 6812-07, San Jose/USA, January 2008.
- [8] Zenzo, S.D.: A Note on the Gradient of a Multi-image. Computer Vision, Graphics, and Image Processing, Vol.33, 1986.
- [9] Efros, A.A., Leung, T.K.: Texture Synthesis by Non-Parametric Sampling. IEEE ICCV, Vol.2, p.1033–1038, 1999.
- [10] Haworth, C., Peacock, A.M., Renshaw, D.S.: Performance of Reference Block Updating Techniques when Tracking with the Block Matching Algorithm. IEEE ICIP, Vol.1, 2001.
- [11] Hays, J., Efros, A.A.: Scene Completion using Millions of Photographs. ACM Trans. Graph., Vol.26 (3), 2007.
- [12] Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.: Image Analogies. SIGGRAPH, Comp. Graph. Proc., 2001.
- [13] Kervrann, C., Boulanger, J.: Optimal spatial adaptation for patch-based image denoising. IEEE Imag. Proc. Vol.15, 2006.
- [14] Kindermann, S., Osher, S., Jones, P.: Deblurring and Denoising of Images by Nonlocal Functionals. SIAM Multiscale Modeling and Simulation, Vol.4, p.1091–1115, 2005.
- [15] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. IEEE PAMI, Vol.12 (7), July 1990.



(a) Color image + noise. (b) Bilateral filtering[19]. (c) Anisotropic PDE[20].



(d) Non-Local Means [5]. (e) Patch-based anisotropic diffusion (Our proposed method Eq.(4)).

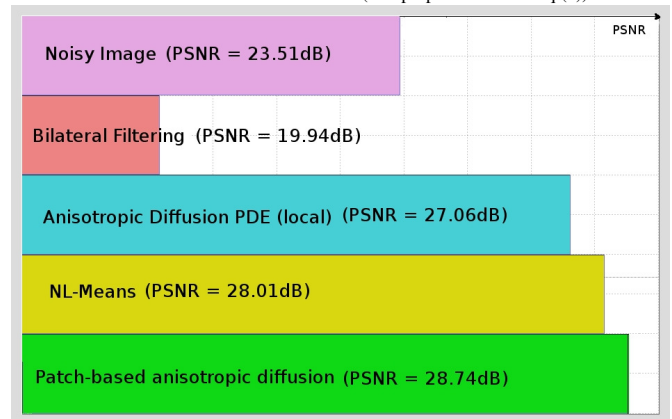


Fig. 2. Comparison of denoising results obtained with the proposed non-local diffusion PDE versus other edge-preserving methods.

- [16] Peyré, G.: Manifold Models for Signals and Images. Research Report, Ceremade, No.2007-14, 2007.
- [17] Rudin, L. Osher, S. Fatemi, E.: Nonlinear Total Variation based noise removal algorithms. Physica D, Vol.60, pp.259-268,1992.
- [18] Sapiro, G.: Geometric Partial Differential Equations and Image Analysis. Cambridge University Press, 2001.
- [19] Tomasi, C., Manduchi, R.: Bilateral Filtering for Gray and Color images. ICCV. p.839–846, 1998.
- [20] Tschumperlé, D., Deriche, R.: Vector-valued Image Regularization with PDE's: A Common Framework for Different Applications. IEEE PAMI, Vol.27 (4), p.506–517, 2005.
- [21] Tschumperlé, D.: Fast Anisotropic Smoothing of Multi-Valued Images using Curvature-Preserving PDE's. IJCV, Vol. 68(1), pp. 65–82, 2006.
- [22] Wei, L.Y., Levoy, M.: Fast Texture Synthesis using Tree-structured Vector Quantization. SIGGRAPH, Computer Graphics Proceedings, p.479–488, 2000.
- [23] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner-Verlag, Stuttgart, 1998.