Low-level fusion: a PDE-based approach

Sorin Pop, Olivier Lavialle, Equipe Signal et Image
LAPS-IMS UMR 5218
Talence, France
pop@enseirb.fr, lavialle@enseirb.fr

Romulus Terebes and Monica Borda
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
Romulus.Terebes@com.utcluj.ro
Monica.Borda@com.utcluj.ro

Abstract - In this paper, we present a new general method for image fusion based on Partial Differential Equation (PDE). We propose to combine pixel-level fusion and diffusion processes through one single powerful equation. The insertion of the relevant information contained in sources is achieved in the fused image by reversing the diffusion process. To solve the well-known instability problem of an inverse diffusion process, we introduce two additional constraints. Then, we propose a general PDE, which integrates one of these constraints as a regularization term. One of the advantages of such an original approach is to improve the quality of the results in case of noisy input images. To answer to the requirements of a 3D specific fusion application, we also propose an extension of the general equation. Finally, few examples and comparisons with classical fusion models will demonstrate the efficiency of our method both on 2D and 3D cases.

Keywords: Image fusion, Partial Differential Equation, anisotropic diffusion, 3D application

1 Introduction

The image fusion is a process which consists in combining different sources to increase the quality of the resulting images. In case of pixel-level fusion, the value of the pixels in the fused image is determined from a set of pixels in each source image.

In order to obtain output images which contain better information, the fusion algorithms must fulfill some requirements: the algorithm must not discard the relevant information contained in the input images; additionally, it must not create any artifacts or inconsistencies in the output images.

In the last decade, many studies were dedicated to image-level fusion methods [1]. Among the classical methods, we can mention the well known methods based on pyramid decompositions [2,3], wavelet transform [4], or different weighted combinations [5]. These techniques were applied in a wide variety of application fields including remote sensing [6], medical imagery [7] or defect detection [8].

The most popular fusion methods are based on a multiscale decomposition. These approaches consist in performing a multiscale transform on each source image to obtain a composite multiscale representation. Then, by defining a selective scheme, the fused image is obtained through the use of an inverse multiscale transform.

In this paper, we propose an original image-level fusion approach based on the use of a Partial Differential Equation. The PDE formulation is inspired by the works dedicated to the non-linear diffusion filters.

Initially proposed by Perona and Malik [9], the non-linear diffusion filters have been widely used in recent years in edge preserving and enhancement filtering. The gray levels of an image \( U \) are diffused according to:

\[
\frac{\partial U}{\partial t} = \text{div}(c(x,y,t)\nabla U)
\]  

(1)

The scalar diffusivity \( c(x,y,t) \) in a pixel of coordinates \( x, y \), chosen as a non-increasing function \( g \) of the gradient \( \nabla U \), governs the behavior of the diffusion process. A typical choice for the diffusivity function \( g \) is [9]:

\[
c(x,y,t) = g(|\nabla U|) = \frac{1}{1 + (|\nabla U|/K)^2}
\]  

(2)

with \( K \) some gradient threshold. Practical implementations of the P-M filter give impressive results, noise is eliminated and edges are kept or even enhanced provided that their gradient value is greater than the threshold \( K \).

By already classical results, (1) can be put in terms of second order derivatives taken in the directions of the gradient vectors (\( \eta \)) and in the orthogonal ones (\( \xi \)):

\[
\frac{\partial U}{\partial t} = g(|\nabla U|)U_{\xi\xi} + g(|\nabla U|)U_{\eta\eta} \quad \text{or} \quad \frac{\partial U}{\partial t} = g(|\nabla U|)U_{\eta\eta} + g(|\nabla U|)U_{\xi\xi}
\]  

(3)

This expression allows an easier interpretation of the original equation, which acts like a low pass filter on the
edge directions and selectively, it can enhance edges approaching a backward diffusion for $|\nabla U| \geq K$.

Since, several drawbacks of the classical diffusion filter were mentioned. In [10] Catté et al. are showing that the P-M filter is ill-posed and can also enhance noise. By simply replacing the original image in the diffusivity function by a Gaussian smoothed one $U_\sigma = G_\sigma \ast U$, the authors establish the existence, uniqueness and regularity of the solution for their improved filter:

$$\frac{\partial U}{\partial t} = \text{div} \{ g \left[ \nabla G_\sigma \ast U \right] \nabla U \}$$

(4)

The regularized anisotropic diffusion equation does not have a directional interpretation; however, from a practical point of view, the authors noticed similar results with the PM filter.

Shock filters constitute another successful class of PDE-based filters. In order to sharpen an image, these filters, initially proposed by Osher and Rudin [11] employ an inverse diffusion equation. The well-known stability problem of the inverse heat equation is solved for the discrete domain by the mean of minmod function [11].

Other important theoretical and practical contributions were brought by Weickert [12-13]. The proposed EED (Edge Enhancing Diffusion) and CED (Coherence Enhancing Diffusion) models are anisotropic diffusion methods or often called tensor based diffusion.

The general equation is written in PDE form, as:

$$\frac{\partial U}{\partial t} = \text{div}(D \nabla U)$$

(5)

with $D$ some square 2*2 matrix for 2-D images and two additional boundaries and initial conditions.

The purpose of a tensor based approach is to steer the smoothing process according to the directional information contained in the image structure. In the CED, the diffusion matrix $D$ is created based on the tensor structure. This tensor is a powerful tool for analyzing coherence structures:

$$J_\rho(\nabla U_\sigma) = K_\rho \ast (\nabla U_\sigma \otimes \nabla U_\sigma)$$

(6)

Each component of the resulted matrix of the tensor product ($\otimes$), is convolved with a Gaussian kernel ($K_\rho$). The eigenvectors of $J_\rho$ represent the average orientation of the gradient vectors and the structure orientation, at scale $\rho$. The diffusion matrix $D$ in (5) has the same eigenvectors as $J_\rho$ but its eigenvalues are chosen according to a coherence measure. The diffusion process acts mainly along the structure direction and becomes stronger when the coherence increases. In this manner, the model is even able to close interrupted lines.

Recently, specific PDE-based approaches were devoted to 3D filtering [14-16]. One of these approaches dedicated to seismic domain will be briefly described in section 2.4.

In the next section, we will introduce a PDE formulation considering the source images as initial states of a diffusion process. We will extract fused versions of the images depending on the temporal evolution of the process. At each step the PDE will take into account the information contained in each source image, leading to a more convenient set of resulting images. In order to ensure the stability of the process, two additional constraints are introduced. To deal with noisy inputs, we propose a fusion-diffusion scheme by adding a diffusion term in the PDE. Finally, an extension for 3D applications in seismic domain is presented.

In section 3, we will show some results obtained by our fusion approach on blurred images and we will compare these results with those provided by some classical approaches. Then we will illustrate the efficiency of our approach in the case of noisy source images. In addition, the proposal equation for seismic application will be tested on the synthetic noisy 3D blocks.

Conclusions and perspectives are given in section 4.

## 2 PDE-based fusion

### 2.1 Fusion term

In pixel-based fusion, we consider that each source image provides a part of the relevant information we want to obtain in the output.

We proposed to apply a PDE-based evolution process for each source image. At each step of the process, we are interested in keeping the relevant information contained in the current source while adding the information provided by each pixel in the other images.

To achieve this task we propose a PDE process involving an inverse diffusion process. The general continuous evolution equation of a source data can be formalized as:

$$\frac{\partial U_i}{\partial t} = -\beta_i \text{div} \left( g \left[ \left| \nabla U \right| \max \right] \nabla U_\max \right)$$

(7)

where $i$ represents the current source, $\max$ denotes the source corresponding to the maximum absolute value of the gradient and $\beta_i$ is a positive weight parameter:

$$\beta_i = \begin{cases} 0 & \text{if } i = \max \\ \beta \in [0:1] & \text{otherwise} \end{cases}$$

(8)

The weight parameter ($\beta$) sets the importance of fusion. Even if equation 7 describes the evolution of a single image ($i$), the principle of our approach is to perform the process on each of the input images. The images are updated in parallel at each time step.

The aim is to inject in the current image the relevant information from the other sources. We consider that in each location the relevant information is provided by the image corresponding to the maximum absolute value of the gradient.

By looking for the maximum of the absolute gradient value, edge detection is performed. We search the
maximum of gradient for each pixel. If the maximum gradient occurs in the current image, the current pixel remains unchanged \( \beta_i = 0 \). Otherwise, if the maximum is detected in another source, we inject the difference observed by inverting a diffusion process. The quantity of the fusion can be modulated by a function \( g_F \) of the absolute gradient value. In this paper we adopt the constant positive function \( g_F(\cdot) = 1 \), which will provide an isotropic behavior for the fusion process. Thus, the fusion process is a linear inverse diffusion process, which is similar to a Gaussian de-convolution. The use of a diffusion equation in a discrete image domain requires an appropriate numerical scheme. We adopt an explicit time scheme and the forward and backward approximations for spatial derivatives. The maximum gradient absolute value is evaluated for the nearest neighborhood (4 pixels for 2D case).

We present the numerical scheme for 1D case with the constant positive function \( g_F(\cdot) = 1 \):

\[
\frac{\partial U}{\partial t} = -\beta \left[ D^i_x(U_{\text{max}}) - D^i_x(U_{\text{max}}) \right] \tag{9}
\]

where

\[
D^i_x(U) = \pm \frac{U(x \pm dx) - U(x)}{dx} \tag{10}
\]

for both terms inside the brackets, \( \text{max} \) denotes the source corresponding to the maximum absolute value of the gradient.

### 2.2 Constraints

The major drawbacks of this type of process are the instability, noise amplification and oscillations [17]. To illustrate these drawbacks, we will first apply the fusion process in the one dimensional case to the signals A and B (Fig. 1).

![Figure 1. Evolution of two 1D signals based only on fusion term (equation 9)](image)

As can be observed in Figure 1, using only the inverse diffusion equation described in (7 and 9) leads to oscillations with increasing amplitudes for both signals.

The minimum and maximum of original signals (first row, \( t=0 \)) are surpassed.

We limit these undesirable effects by imposing the boundaries for the amplitude of each sample (gray level of pixel in 2D or voxels in 3D):

\[
\min(U_i^{t=0}) \leq U_i \leq \max(U_i^{t=0}) \tag{C1}
\]

The limits are fixed considering maximum and minimum values through all sources and are applied at each time step.

Thus the oscillations are limited between the minimum and maximum for each sample (Fig. 2).

![Figure 2. Evolution of two 1D signals limited by the C1](image)

So, the C1 constraint limits the oscillations and maintains the outputs in the dynamic range of the inputs (the minimum-maximum principle).

In addition, we are interested to avoid any oscillations appearing in the previous 1D example; the aim is to obtain at the end of the process a ‘Signal A’ identical to input ‘Signal B’ and to preserve the ‘Signal B’.

To solve this problem, we impose a second constraint by forcing the difference between two neighboring pixels to be limited by the maximum of the difference observed in the input images (neighborhood constraint). For 1D case, this can be written as:

\[
\begin{align*}
\min_k [D^i_x(U_i^{t=0})] &\leq D^i_x(U_i) \leq \max_k [D^i_x(U_i^{t=0})] \\
\min_k [D^i_x(U_i^{t=0})] &\leq D^i_x(U_i) \leq \max_k [D^i_x(U_i^{t=0})]
\end{align*} \tag{C2}
\]

Where \( U_i \) with \( 1 \leq k \leq K \) is the \( k \)-th of the \( K \) inputs. Precisely, (C2) forces the value of each difference between two neighbors to be within two bounds: the lower bound is negative (if there exists a negative step in the inputs) or null and the upper bound is positive (if there exists a positive step in the inputs) or null.
For 2D case, two other limits corresponding to North and South differences are added to the East and West differences of 1D case.

We limit the current sample to the minimum or maximum values according to C2. Using this second constraint, no new local edge is created. We reach the desired states for the 1D signals in a short time (Fig. 3). Unfortunately, from a practical point of view the result depends on the order of computation. To avoid this problem, we propose to integrate C2 constraint into the partial differential equation as a term of regularization:

$$\frac{\partial U}{\partial t} = -\beta \text{div}(g_R(\nabla U)_{\max}) - \gamma \text{div}(g_R(\nabla U, \nabla U^i_{\max})_{\nabla U})$$

where $\gamma$ is a positive weight regularization parameter, which sets the importance of the regularization term and $g_R$ is a function which is different from zero when the constraint C2 is not respected. $g_R$ will be defined in (13) for the discrete version of the PDE.

The equation 12 presents the discrete version of the equation 11.

So, it consists in minimizing the differences between the gradient at time $t$ and the maximum gradient at $t=0$. If the maximum (respectively minimum) gradient value at $t=0$ is greater (respectively less) than 0, this value is considered as the upper (respectively lower) limit for the actual gradient value.

Figure 4 shows the results obtained with equation 12.

$$\text{with } g_R \text{ function :}$$

$$g_R(D^i(U), D^i(U^i_{\max})) = \begin{cases} 
D^i(U) - \min_k [D^i(U^i_{max})] & \text{if } D^i(U) < \min_k [D^i(U^i_{max})] \\
D^i(U) - \max_k [D^i(U^i_{max})] & \text{if } D^i(U) > \max_k [D^i(U^i_{max})] \\
0 & \text{otherwise}
\end{cases}$$

A study of the influence of $\gamma$ on the convergence of the process will be the subject of a further work.

The transitions of the impulsion in ‘Signal B’ are injected in ‘Signal A’ by the means of fusion term described above. But the flat zone between the 13th and 17th samples are obtained in ‘Signal A’ after a time $t=4.8$, by the means of regularization term. Cause of the fusion term for the flat zone, ‘Signal B’ tries to follow ‘Signal A’ (‘Signal A’ presents a high gradient value detected by the fusion term and injected in ‘Signal B’). The time of convergence depends on the width of impulsion and on the weight regularization parameter ($\gamma$).

In the frequency domain, the flat zones are characterized by the low frequency. This regularization term can be viewed as the fusion of low frequency.

Note that, the time step $dt$ was set at 0.1 and $g_R=1$ for all 1D experiments, while $\gamma=1$ for the last example.

The 1D discrete model described in (12) can be easily extended for 2D (respectively 3D) data by considering a 4-neighborhood (respectively 6-neighborhood).

Contrary to the classical fusion methods, our algorithm provides one output for each source signal. Obviously, the aim is to obtain similar outputs while the relevant information is preserved. In practice, we can observe a convergence of the process: the distance (i.e. RMSE) between the fused images decreases in time. The stopping time, like in diffusion case, is chosen by the human operator; nevertheless a criterion based on a distance measure or a quality factor calculation can be proposed.
2.3 Diffusion term

One of the benefits of our model is the possibility to add a denoising process during the fusion process. This denoising process can be achieved by adding another term to equation 7. As diffusion term, for the 2D case, we adopt the Catté model [10]. The equation 7 becomes:

\[
\frac{\partial U_i}{\partial t} = \text{div}(g \nabla U_i) \nabla U_i - \beta_i \text{div}(g \nabla U_{\max}) \nabla U_{\max} \quad (14)
\]

where only the C1 constraint is added.

In classical approaches the noise is detected as relevant information and is injected in the fused images. Thus the obtaining of a noise-free output image requires a preprocessing step for denoising the input data. In the presence of noise the maximum and minimum of the first constraints (C1) are evaluated at each time step. In this way, the noise at t=0 is not taken into account. We apply the same principle if the C2 constraint is employed.

2.4 Diffusion term for 3D data. Application on seismic data

The above framework can be easily applied for 3D data. For instance, one application field for 3D fusion data is the seismic domain.

In this case, the aim is to combine the seismic information provide by different sources (i.e. acquisition for different azimuths leading to different seismic 3D blocks representing the same subsol area). First of all, the 3D blocks are registered. Then a fusion-diffusion process is used to denoise the 3D blocks, while the seismic faults are preserved and injected in the fused blocks. So, the seismic faults which are recognized by the discontinuities of seismic horizons are the key information for this type of data.

As diffusion process, we adopt a dedicated seismic diffusion: Seismic Fault Preserving Diffusion (SFPD) [16], but other models can be considered as well. SFPD is a 3D extended model based on Weikert’s 2D CED diffusion. The general 2D model (15) becomes in 3D:

\[
\frac{\partial U_i}{\partial t} = \text{div}(D \nabla U_i) - \beta_i \text{div}(g \nabla U_{\max}) \nabla U_{\max} + \gamma \text{div}(g \nabla U_i) \nabla U_i \quad (16)
\]

where \( D \) is a 3x3 matrix, \( \lambda \) takes different values, \( \alpha \) is a constant and \( k \) is a coherence measure.

The diffusion 3*3 matrix \( D_i \), specific at each source, has the same eigenvectors as the structure tensor \( J_p \) (equation 6), but its eigenvalues are chosen according to a seismic confidence measure \( C_{\text{fault}} \), introduced by Bakker [18]. The objective of \( C_{\text{fault}} \) is to discriminate between fault neighborhoods and non-broken horizons:

\[
C_{\text{fault}} = \frac{\mu_2 - \mu_3}{\mu_2 + \mu_3} \quad (17)
\]

where \( \mu_1 > \mu_2 > \mu_3 \) are the eigenvalues of \( J_p \). In order to denoise and preserve the seismic faults, we employ this measure in the system of choosing the matrix D eigenvalues:

\[
\lambda_1 = \alpha \quad \text{if } k = 0
\]

\[
\lambda_2 = \lambda_3 - (\lambda_3 - \lambda_1) \delta_i(C_{\text{fault}})
\]

and

\[
\lambda_3 = \alpha + (1 - \alpha) \exp\left( -\frac{C}{k} \right) \quad \text{else}
\]

where \( \alpha \), \( C \) are constants and \( k \) is the coherence measure proposed by Weickert. \( h, f \) is an increasing sigmoid function described in [19] taking values in \([0 ; 1]\) which allows to parameterize the influence of the confidence measure. The originality of our system lies in the second eigenvalue. \( \lambda_2 \) takes values between \( \lambda_1 \) and \( \lambda_3 \) and depends continuously on the measure \( C_{\text{fault}} \).

The system allows diffusion in only one orientation for the case of fault neighborhoods while for the layers case it will diffuse guided by two orientations. Thus we are able to denoise the seismic blocks while the faults are preserving. Also, this approach exempts from the creation of false anisotropic structures, artifacts typically observable in images processed with the classical tensorial models [16].

3 Results

3.1 2D applications

We choose to examine the efficiency of our 2D model in an out-of-focus image problem. Figures 5a and 5e show the details of two known images with different zones of focus. We present in Figure 5(d,h) the corresponding fused images obtained after 200 iterations with a time step \( dt=0.1 \) and a weight parameter \( \beta=1 \). We have employed the discrete equation 9 with the two constraints described in section 2.2. Let us compare the results provided by our method with some classical fusion scheme results. Among the classical fusion methods implemented in the free Matlab tool: fusetool conceived by Rockinger, we evaluate the Laplacian (LAP) pyramid method and the Shift Invariant Discrete Wavelet Transform (SIDTW) method (with Haar function) [4].
Figures 5c and 5g illustrate the results obtained after 6 decomposition levels for Laplacian pyramid respectively 3 decomposition levels in the case of SIDTW. In both cases, the choose-max selection scheme was applied for the high-pass combination and the average of inputs for the low-pass combination. For a visual comparison, we also present the results obtained with the PCA (Principal Component Analysis) method (Fig. 5f) and by averaging the inputs (Fig. 5b).

For a quantitative comparison of the fusion methods, we adopt the weighted fusion quality measure proposed by Piella [22]:

$$Q_W(u,v,f) = \sum_w c(w) [\lambda_u(w) \cdot Q(u,f | w) + \lambda_v(w) \cdot Q(v,f | w)]$$

where $Q$ is the Wang and Bovik quality factor [23] computed in the window $w$.

The Wang and Bovik quality factor quantifies the structural distortion between two images. It is composed of three factors: correlation, distortion of mean luminance and distortion of contrast:

$$Q(u,v) = \frac{\sigma_u}{\sigma_u + \sigma_v} \cdot \frac{2 \sigma_u \sigma_v}{\sigma_u^2 + \sigma_v^2}$$

$\sigma_u^2$ and $\sigma_v$ stand for variance respectively covariance and $\mu$ denotes the mean luminance of $u$.

In (19) $\lambda_u(w)$ quantifies the importance of input $u$ relative to input $v$; $c(w)$ is the overall saliency of a window. These measures employ salience information such as variance, entropy, contrast or gradient norm.

We chose the variance computed in a 7 by 7 pixel window as salient information. The variance acts as edge detector, which is a desirable result in this specific fusion problem.

Note that the quality factor was computed on the detail images.

The PCA and the results obtained by averaging have low quality factor $Q_W=0.867$, respectively $Q_W=0.869$, which reflects the poor visual quality. Our proposed approach obtained a similar quality factor ($Q_W=0.941$) as the LAP pyramid method ($Q_W=0.941$) or SIDTW method ($Q_W=0.942$). Among fusetool methods these last two methods provide the best results for this application.

In addition a visual comparison certifies that our results are comparable with the images produced by the best fusion methods. The in-focus zones are detected by the absolute gradient value and are injected by the inverse diffusion equation into the output images. The high quality factor certifies that the saliency information (edges here) is well injected from the inputs into the output images.

To quantify the similarities between two output images we use the root-mean-square error (RMSE):

$$RMSE(U_A,U_B) = \sqrt{\frac{\sum_{x,y} (U_A(x,y) - U_B(x,y))^2}{n}}$$

where $n$ denotes the total number of pixels.

We observe that the RMSE has a powerful decreasing slope. The RMSE between the input images is equal to 14.80 and it is drastically reduced to 0.41 at the end of process. Thus, the output images are quite similar. In order to have one single output image, at the end of process, the average of the outputs or a simple selection based on quality factor are possible.

To illustrate the efficiency of our approach in case of noisy inputs, we added to the original out-of-focus images a Gaussian noise of $\sigma_0 = 15$ (to obtain a signal-to-noise
ratio SNR=9db for both images). In Figure 6 we present the noisy input images as well as the fused images.

![Figure 6 a,c) The noised input images; b,d) fused images](image)

The fused images are obtained with equation 14 and the two constraints after 100 iterations with a time step $dt=0.1$ and a weight parameter $\beta=1$. In addition, parameters specific to diffusion are set to $\sigma=0.8$ and $K=2.5$ as threshold. The RMSE is reduced from 25.01 to 2.40 at the end of the process.

As it can be observed, the noise is discarded from the input images while the in-focus zones are well injected and preserved in the output images.

In the case when we dispose of non-noisy input images the weight fusion quality factor can be useful to discriminate between the outputs. But in the real noisy cases, such a quality factor is tributary to the saliency measure, which incorporates the noise as well as the pertinent information. A noisy-free saliency measure is also an object of our studies.

The 2D approach developed in this paper can be easily extended using another anisotropic diffusion equation.

### 3.2 3D seismic application

In this section, we present the results obtained by the seismic 3D model (subsection 2.4). Since it is much easier to judge the efficiency of the fusion process on a synthetic image, we propose to use 3-D synthetic blocks composed by a stack of layers with a sinusoidal profile. One of them is clearly broken by a fault. In the second the fault was smoothed using a frequency filtering. Both of them are corrupted with additive Gaussian white noise ($\sigma_w=25$). Figure 7(a,c) shows a front section of the noisy blocks (128*124*129 voxels).

The fusion-diffusion process applied to the noisy blocks leads respectively to the blocks shown in Figure 7(b,d).

![Figure 7 a,c) The noisy synthetic blocks (front section); b,d) output blocks](image)

We have employed the general 3D equation (16). The results are obtained after 50 iterations with a time step $dt=0.1$, a weight parameter $\beta=1$ for fusion term. We have set the weight regularization parameter at 1 ($\rho=1$). In addition, the parameters specific to SFPD diffusion are set to $\alpha=10^{-6}$, $\sigma=0.4$, $\rho=1.2$ and $\tau=0.05$ as threshold for sigmoid function.

The efficiency of the diffusion process is clearly illustrated: the two blocks are filtered and the fault is preserved in the block in first row. The fusion process allowed to inject the fault in the block in the second row. RMSE between the input blocks is equal to 36.11 and it is equal to 0.018 between the output blocks. Thus, the output blocks are similar.

### 4 Conclusions and perspectives

In this paper we propose a new approach for image fusion based on an inverse diffusion process. The proposed formulation allows to deal with noisy inputs through the use of a diffusion process along with the fusion process. The advantage of such an approach lies in the possibility to adapt the fusion and diffusion processes to different types of applications.

In the further works we would like to propose an optimal stop criterion for the process. In addition, we will focus on finding different powerful anisotropic functions for fusion ($g_F$). Finally, a study on the convergence of the outputs will be carried out.

### References

