

Partial differential equation-based digital image compression models

Tudor Barbu

Abstract. The state of the art partial differential equation (PDE)-based image compression techniques are surveyed in this research article. An overview of the image coding approaches is described first. Next, the most important PDE-based models used in the decompression stage are presented here. Thus, image decompression schemes based on linear homogeneous diffusion of various orders, nonlinear anisotropic diffusion and edge-enhancing diffusion (EED), and using encoding algorithms such as B-tree triangular coding (BTTC), rectangular subdivision or edge-based coding for image sparsification are discussed in this survey. Also, our own contributions in this domain, representing effective compression and decompression solutions using PDE-based edge detection and nonlinear anisotropic diffusion-based inpainting, are described in this paper and compared with the state of the art techniques.

M.S.C. 2010: 35Kxx, 60G35, 65L12, 68U10, 68Pxx, 68P30, 68Txx, 94A08.

Key words: image compression; coding algorithms; image sparsification; partial differential equations; linear homogeneous diffusion; nonlinear diffusion models; PDE-based inpainting; edge-enhancing diffusion; numerical approximation scheme.

1 Introduction

Digital image compression represents an important sub-domain of both the image processing and data compression fields. The purpose of a compression task is to reduce the size of the image file without losing much information and maintaining its visual quality, in order to facilitate the storage and transmission processes. The image content is encoded using fewer bits than its original representation, in the compression process that can be either lossless or lossy [14].

The lossless image compression methods remove or reduce the statistical redundancy, and recover perfectly the image at decompression, no information being lost. The following coding algorithms are used for lossless compression: the Huffman coding, Arithmetic coding, Run Length Encoding (RLE), LZW encoding and Area coding [7, 14, 24]. Several image formats, like BMP, GIF or PNG are based on the lossless compression.

The lossy coding algorithms produce much higher compression rates, but also lose some of the image information. While the decompressed images are not identical to the originals, they are similar enough. The lossy encoding methods include vector quantization, transformation-based coding, fractal coding, Block Truncation Coding and sub-band coding [7, 24]. Well-known image standards based on lossy compression include JPEG, JPEG 2008 and other versions of it, which use transform-based (DCT and DWT) image encoding [7, 14, 24], and the MPEG format with its variants [7, 24].

Image compression approaches using partial differential equation (PDE)-based models represents a recently developed class of lossy compression schemes. In the last 30 years, the PDEs have been widely applied in various static and video image processing and analysis sub-domains, such as image filtering [2], inpainting [27], segmentation [21], registration [29], compression [23] and video motion estimation [18].

We have performed a lot of research in the PDE-based image processing and analysis fields, and elaborated numerous variational and PDE models for image denoising [6], interpolation [5], segmentation [3] and compression [4]. We approach here the digital image compression domain, providing an overview of the state of the art PDE-based compression techniques.

The PDE-based image compression could be considered an application area of the PDE-based inpainting, since those compression methods make use of the PDE image completion models for the decompression task. The differential models are rarely used in the compression stage, usually being used in the pre-processing steps that enhance the image for the compression task. Then, the image data compression is performed by applying some image sparsification and coding procedures. Some sparsification and encoding algorithms that are successfully used together with PDE models are surveyed in the next section, where the general form of the PDE-based decompression problem is also presented.

Then, the state of the art partial differential equation-based image compression and decompression models, such as those based on nonlinear anisotropic diffusion and edge-enhancing diffusion (EED), are described in the third section. While these techniques use the PDEs in the decompression stage only, the nonlinear diffusion-based image compression framework proposed by us is using PDE models for both compression and decompression tasks. Our own contribution in this domain is briefly described in the fourth section and compared to the state of the art compression solution. Conclusions of this article are drawn in a final section and the paper ends with a section of references.

2 Image coding solutions and PDE-based decompression

In the image compression process stage one have to select a subset of the image pixels, which are then coded and stored according to an algorithm. So, in the first step, a sparsification is performed on the digital image.

The idea is to reduce the image data to a well-adapted set of significant sparse points that can be then coded efficiently. Many image sparsification and coding solutions that work properly with the PDE inpainting models used in the decompression stage have been developed.

The *random sparsification* approach selects randomly the scattered data points from the image, which are then encoded. Unfortunately, this image sparsification solution may produce decompressed images of poor quality and a high loss of information.

A much better image encoding solution is *B-Tree Triangular Coding (BTTC)* introduced by R. Distasi et al. in 1997 [11, 23]. It creates an useful sparse point representation and an efficient coding of the sparsified image data. Their coding scheme is based on the recursive decomposition of the image domain into right-angled triangles arranged in a binary tree. The BTTC domain triangulation for *Lenna* image at 1.20 bpp is described in Fig. 1.

BTTC represents a fast encoding technique, is characterized by a quite convenient computational cost $O(n \log n)$ for a number of pixels n , and is quite easy to implement and highly parallel. The compression technique described in [11] is based on a linear interpolation of all these triangles in the image decoding stage. This binary tree-based coding can be also used successfully together with the PDE-based inpainting models. Thus, the scattered data obtained by BTTC could be properly interpolated using PDE schemes in the decompression step, the respective techniques being described in the following sections.

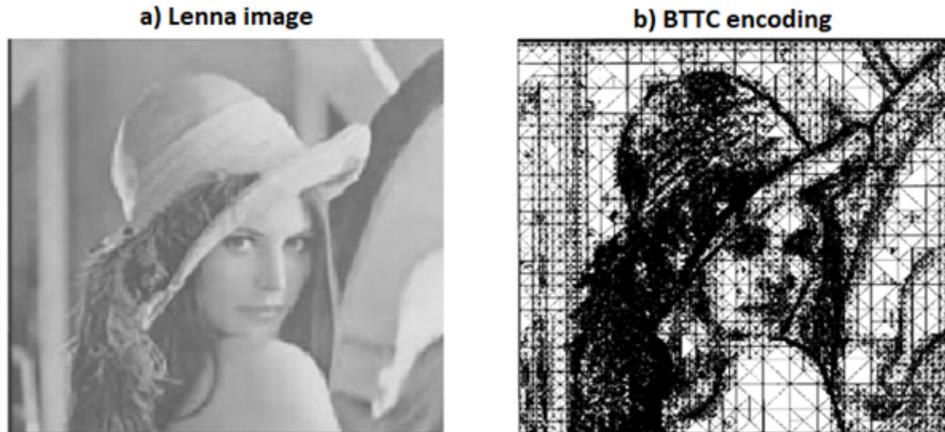


Figure 1: BTTC triangulation of *Lenna* image

Other tree-based sparsification and coding approaches have been also developed. Let us mention here the *image coding method using adaptive Delaunay triangulation*, elaborated by Demaret et al. in 2005 [10]. The *tree-based rectangular subdivision scheme*, which partitions the original image into rectangular subimages, represents another image sparsification solution that is used along some PDE-based interpolation models that are described in the next section [23]. Another effective binary tree-based sparsification is the *stochastic tree-based image sparsification* [23].

The edge-based image coding models has been also used successfully with the PDE inpainting schemes. Such an encoding technique extracts the image boundaries, by performing an edge detection operation, first. Then, only the pixels that are related to the detected edges (positioned on them or in their vicinity) are considered as sparse

points. Next, these selected pixels are coded by using various encoding schemes: Huffman encoding, Run Length Encoding, Arithmetic coding and others. Some PDE-based image compression algorithms using edge-based encoding are discussed in the following section [7, 19].

Clustering-based quantization solutions for PDE-based image compression have been also developed [15]. These quantization-based image encoding approaches reduce the number of colors without altering the quality of image. Unsupervised clustering algorithms like K -means could be used for this purpose. The key pixels corresponding to those detected colors are then selected, coded and stored [15].

The scattered data obtained and coded using these sparsification and coding algorithms is then decoded and interpolated in the decompression stage. Variational and PDE-based inpainting techniques could be applied on the sparse pixels [2, 27]. The PDE compression models achieved high data compression performance measure values: the compression rate (in bits per pixel), compression ratio, fidelity and quality.

Energy-based, or variational, structural inpainting techniques reconstruct the damaged image by solving a minimization problem involving an energy functional composed of a fidelity term and a regularizing term:

$$(2.1) \quad \min_{u \in U} \left(J(u) = R(u) + \frac{1}{2} \int_{\Omega} \lambda_D (u - u_0)^2 d\Omega \right),$$

where D is the inpainting domain $\lambda_D = \lambda \cdot \mathbf{1}_{\Omega \setminus D}$, $\lambda > 0$, the regularizing term $R(u)$ contains certain a-priori information from u and perform the filling, while the fidelity term forces the minimizer u to remain close enough to the initial image outside of the inpainting region [2, 27]. This variational scheme could be applied by using the sparsified image as observed image u_0 , for the decompression task.

Some PDE inpainting models follow the variational principles, being derived from such variational schemes, by applying the Euler Lagrange equations and then the steepest gradient descent method [2]. Other PDE-based inpainting schemes are non-variational, being directly provided as evolutionary equations.

Thus, the general form of a variational PDE-based interpolation model used for image decompression is obtained as follows:

$$(2.2) \quad \frac{\partial u}{\partial t} = (1 - \lambda_D)L(u) - \lambda_D(u - u_0),$$

where $L(u)$ is a differential operator that may take various forms.

The interpolation result is obtained from (2.2) by solving numerically this partial differential equation using an iterative numerical approximation algorithm. The state of the evolving image u obtained at the final iteration represents the decompressed image.

Thus, various PDE-based image compression techniques are obtained by considering variants of the differential operator L . Some of them, representing state of the art compression frameworks, are described in the following sections.

3 State of the art diffusion-based image compression models

Several PDE-based image compression techniques are described in this section. They use linear and nonlinear diffusion-based models.

3.1 Linear homogeneous diffusion-based image compression

A well-known linear homogeneous diffusion-based image compression technique is based on the Harmonic Inpainting, which represents a variational interpolation model. Its differential operator has the following form:

$$(3.1) \quad L(u) = -\nabla^2 u = -\Delta u.$$

Harmonic Inpainting interpolates successfully the sparsified image in the decompression stage, the scattered data points being obtained by using the sparsification methods described in the previous section. This type of compression provides effective results for cartoon image compression and depth map compression [9, 19].

Such a linear homogeneous diffusion-based image compression method is that introduced by M. Mainberger and J. Weickert in 2009 [19]. It uses an edge-based encoding algorithm for image compression.

An edge detection process, based on the Marr-Hildreth operator [20], is performed first. Then, the pixels from the neighborhood of the detected edges are encoded, by applying some quantisation, subsampling and PAQ coding algorithms and stored using the JBIG standard [19]. The image decompression is performed by applying the linear homogeneous diffusion-based inpainting given by (2.2) and (3.1).

This edge-based compression framework provides very good results for cartoon image decompression. It outperforms the JPEG and JPEG 2000 compression standards when applied on cartoon-like images and measured at the same compression rate. One can see an image compression example using edge-based coding and harmonic interpolation in the next figure [19].

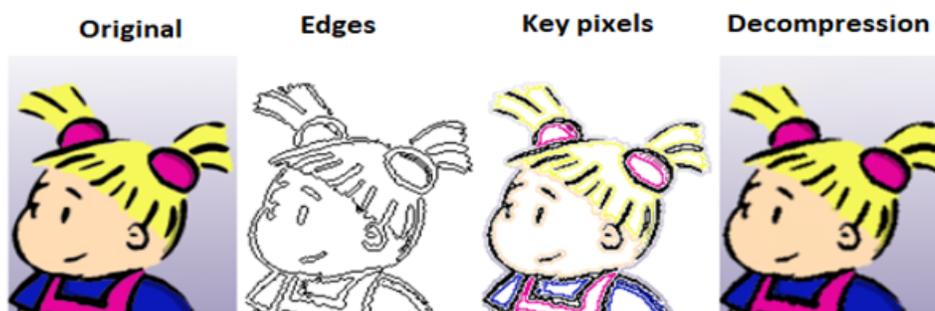


Figure 2: Edge-based compression of a cartoon image

Some extensions of this harmonic operator based image compression model use differential operators of higher orders. Such a compression algorithm is based on the biharmonic smoothing operator [16, 26] having the form:

$$(3.2) \quad L(u) = -\nabla^4 u = -\Delta^2 u.$$

The biharmonic-based image compression technique provides satisfactory decompression results, by applying the scattered data interpolation solution provided by (2.2) and (3.2), when used together with an effective image coding scheme, such as B-Tree Triangular Coding. It produces weaker results when used along random sparsification. An example of image decompression result produced by BTTC+Biharmonic compression technique is described in Fig. 3.



Figure 3: BTTC+Biharmonic image compression

Another linear diffusion-based compression technique derived from harmonic operator based approach is based on triharmonic smoothing operator [26]. That is, a higher-order differential operator having the form:

$$(3.3) \quad L(u) = \Delta^3 u.$$

Triharmonic inpainting given by (2.2) and (3.3) also provides good image decompression results when applied on the sparse pixels encoded by using BTTC. These linear PDE-based compression models of various orders are solved numerically by constructing some finite difference-based numerical approximation schemes based on the Laplacian discretization [17].

3.2 The absolute monotone Lipschitz extension (AMLE) model

A second-order PDE-based interpolation technique that has been successfully used for image compression is the absolute monotone Lipschitz extension (AMLE) model [1, 26]. This partial differential equation-based inpainting algorithm is based on the second order directional derivative in the gradient direction.

Thus, the AMLE-based compression model is characterized by the following differential operator:

$$(3.4) \quad L(u) = \partial_{\eta\eta} u,$$

where

$$(3.5) \quad \eta = \frac{\nabla u}{|\nabla u|}.$$

The AMLE inpainting approach given by equations (2.2), (3.4) and (3.5) achieves very good decompression results when applied on a properly selected scattered data point set, obtained for example with a B-tree based coding. It provides a weaker decompression output when applied on a randomly selected set of sparse pixels, as one can see in the Lenna image compression example described in Fig. 4.

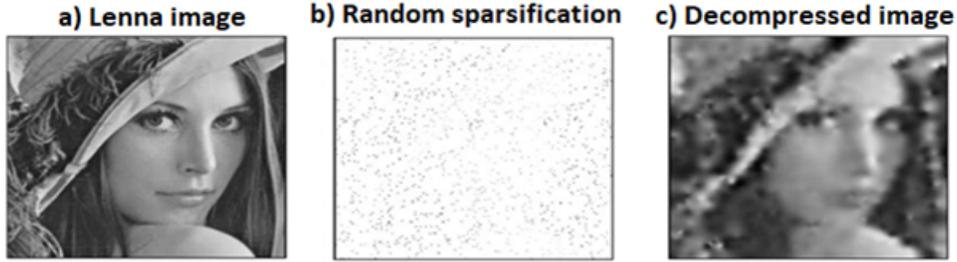


Figure 4: AMLE-based image compression example

3.3 Nonlinear Anisotropic Diffusion-based Image Compression

The second-order nonlinear anisotropic diffusion models have been successfully applied in the image restoration and inpainting domains [28]. Therefore, they have been also introduced in the digital image compression field, since they represent powerful interpolation solutions for the scattered image data [13, 28]. Nonlinear second-order anisotropic diffusion-based image compression is characterized by the next class of smoothing operators to be used in (2.2):

$$(3.6) \quad L(u) = \operatorname{div}(g(\|\nabla u\|^2)\nabla u),$$

where the diffusivity (edge-stopping) function g is positive, monotonically decreasing and converges to zero.

Each anisotropic diffusion model that can be used in the decompression stage is characterized by such a diffusivity function. Thus, the anisotropic diffusion model of Perona and Malik is characterized by the next two edge-stopping functions [28]:

$$(3.7) \quad g(s^2) = e^{-\frac{s^2}{k^2}}; \quad g(s^2) = \frac{1}{1 + \left(\frac{s}{k}\right)^2}.$$

The *Charbonnier diffusion* scheme is characterized by the function [13, 28]:

$$(3.8) \quad g(s^2) = \left(1 + \frac{s^2}{k^2}\right)^{-1/2}.$$

The robust anisotropic diffusion (RAD) model introduced by Black et al. [6, 28] has the following diffusivity function:

$$(3.9) \quad g(s^2) = \begin{cases} \left(1 - \frac{s^2}{5k^2}\right)^2, & \text{if } \frac{s^2}{5} \leq k^2 \\ 0, & \text{if } \frac{s^2}{5} > k^2 \end{cases}$$

These nonlinear anisotropic diffusion-based interpolation methods provide effective decompression results. They are discretized by using finite difference method-based numerical approximation schemes [17].

3.4 Edge-enhancing diffusion-based compression frameworks

A category of more performant nonlinear PDE-based image compression techniques is based on edge-enhancing diffusion (EED) models. These models provide a powerful nonlinear diffusion-based smoothing. The edge enhancing diffusion preserves and enhances the image boundaries, since it reduces the diffusion across edges and permits it along them.

So, such an effective edge-enhancing anisotropic diffusion-based image encoder was proposed by Garlic, Weickert et. al in 2005 [13]. Their BTTC-EED framework applies an improved BTTC scheme for image compression and EED for decompression. Thus, in the compression stage an adaptive B-tree triangular coding (BTTC)-based sparsification is applied to the image to create the scattered interpolation pixels. The image is decomposed into a number of isosceles triangular regions such that within each region it can be properly recovered by interpolation from the vertices. This triangle-based decomposition is then stored into a binary tree structure [12, 13].

The structure of the tree is stored by traversing it (pre-order or level-order) and storing it one bit per node: a 1 for a node that has children, and a 0 for a leaf. A vertex mask is obtained. To code the grey values in all vertices, one first zig-zag traverses the sparse image created with the binary tree structure and store it in a sequence that is encoded with Huffman algorithm. The coding is further enhanced with a final quantization [13].

Image decompression process is performed in two main steps: the decoding and interpolation [25]. First, the vertex mask is recovered from the described encoding and the sparse image is thus achieved. Next, the scattered data interpolation is performed by using the edge-enhancing diffusion scheme [12, 13].

So, this BTTC-EED approach replaces the linear interpolation of the BTTC-L model [11] to a edge enhancing diffusion-based inpainting procedure that is applied within each triangle from the sparsified image [13]. The EED-based image interpolation is achieved by applying the following diffusion operator to the equation (2.2):

$$(3.10) \quad L(u) := \nabla \cdot (g(\nabla u_\sigma u_\sigma^T) \nabla u),$$

where $u_\sigma = u * G_\sigma$ and G_σ represents a 2D Gaussian filter kernel characterized by the standard deviation σ . The diffusivity function g is properly selected: positive, monotonic decreasing and converging to 0.

BTTC-EED model provides an effective sparse image interpolation, outperforming the previously developed PDE-based inpainting techniques in the decompression stage. It provides better values of the performance metrics, lower *average absolute error* (AAE) and the *mean squared error* (MSE), than harmonic, biharmonic, triharmonic, AMLE and Charbonnier diffusion, which mean better quality of the decompression result. BTTC-EED compression algorithm outperforms also the JPEG compression standard, when compared at the same high compression rate, but it is inferior to JPEG 2000 coding [12, 13].

Some BTTC-EED image compression results (adaptive sparsifications and decompressed images) achieved at several compression rates are displayed in Fig. 5. Also, the method comparison results (AAE and MSE values) illustrating the effectiveness of this compression technique are registered in Table 1.

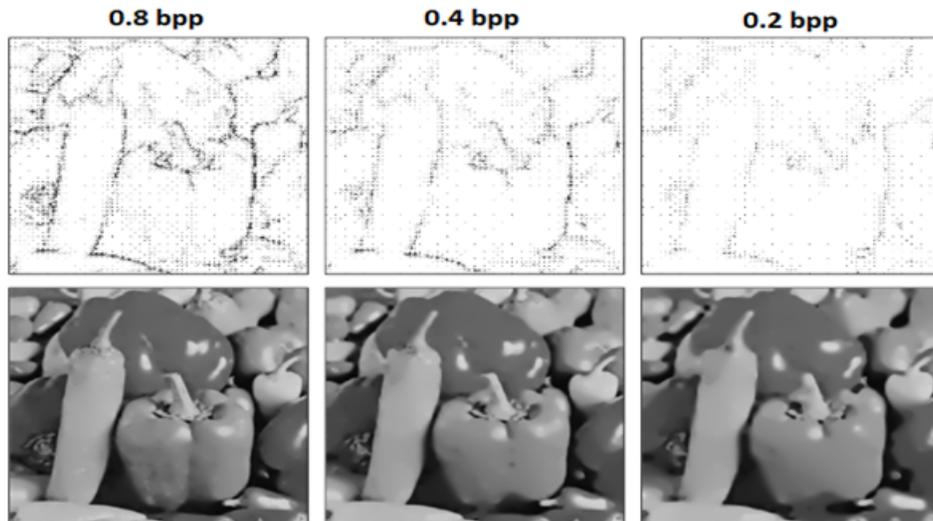


Figure 5: BTTC-EED image compression at several compression rates

Table 1. Method comparison: AAE and MSE values

PDE-based compression technique	AAE	MSE
Harmonic smoothing	16.98	611.5
Biharmonic smoothing	15.79	615.5
Triharmonic smoothing	18.69	807.9
AMLE model	17.33	631.7
Charbonnier diffusion	21.80	987.0
BTTC-EED technique	14.58	591.7

Some improved versions of this edge-enhancing diffusion-based compression approach have also been developed. Such an improved EED-based technique is the Q64+BTTC(L)-EED image codec introduced by Galic et al. in 2008 [12].

An even more effective EED-based compression framework, which is the *Rectangular subdivision with edge-enhancing diffusion (R-EED)* image codec, was proposed by Schmaltz et al. in 2009 [25]. Their technique performs a rectangular subdivision process on the image in the compression stage.

In this subdivision approach a line is saved by using 3 points. Each time one checks the quality of the image reconstruction: if the MSE value is greater than a certain threshold, then the image is split into two sub-images, by saving a line between them. These sub-images are then saved in a recursive manner. The saved pixels values are quantized and coded using an entropy coding algorithm, such as the Huffman coder, Lempel-Ziv-Welch, arithmetic coding and PAQ [28]. The rectangular subdivision process applied to the *Trui* image is described in Fig. 6.

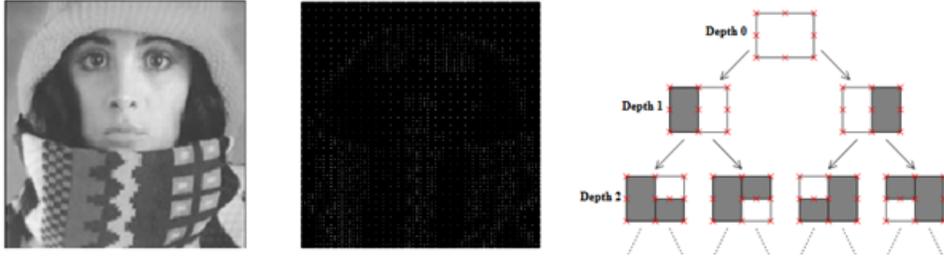


Figure 6: Rectangular subdivision process example

The R-EED decompression is performed by applying an edge-enhancing diffusion-based inpainting technique. So, the differential operator of EED, $L(u) = \text{div}(g(\nabla u_\sigma u_\sigma^T) \nabla u)$, is applied with a Charbonnier diffusivity function, $g(s^2) = \sqrt{\frac{1}{1 + \frac{s^2}{\lambda^2}}}$. An optimization is also performed on this diffusivity function, by selecting an optimal λ parameter that is dependent on the image and the compression ratio [25].

R-EED based image compression clearly outperforms BTTC-EED scheme and all other PDEbased compression methods, and also JPEG. It also provides better decompression results than JPEG 2000 standard, when compared at comparable high compression ratios. While R-EED provides better results for gray-level images, it is outperformed by JPEG 2000 for color image compression. Some method comparison results are described in Fig. 7 that displays the decompression results and MSE values achieved by the EED-based and JPGEg codecs at various compression ratios.

An improved R-EED compression method for color images was introduced by P. Pascal and J. Weickert in 2014 [22]. Their colour compression codec give more attention to the encoding of luma channel of the YCbCr colorspace. This image encoder outperforms JPEG 2000 standard for high compression rates.

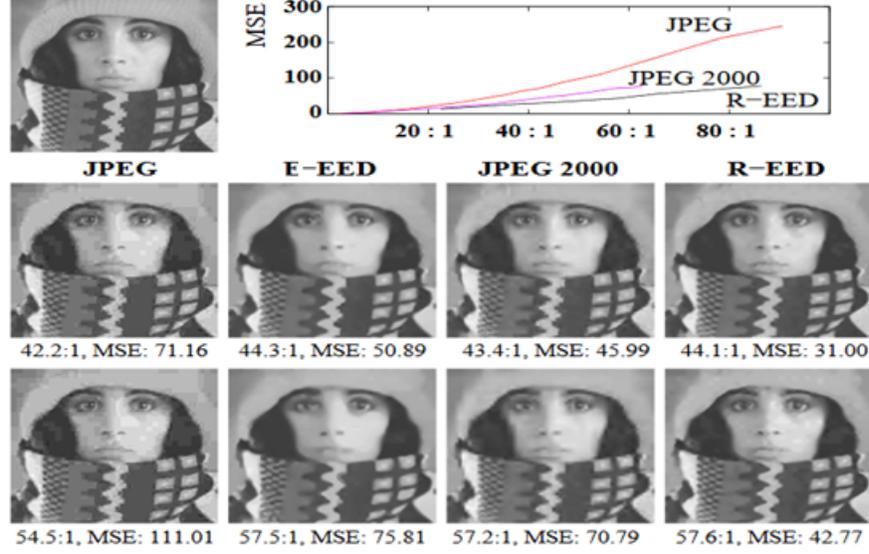


Figure 7: Compression results of various techniques and method comparison

4 Edge-based compression using nonlinear diffusion-based models

Our own contributions in the partial differential equation-based image compression domain are described briefly in this section. So, we have developed recently a nonlinear PDE-based compression technique using edge information, which is disseminated in [4]. Unlike other PDE-based approaches, it uses nonlinear anisotropic diffusion schemes for performing both the image compression and decompression tasks.

Thus, an edge-based image compression technique is proposed in [4]. A second-order nonlinear diffusion-based edge detection is performed first, using the PDE model:

$$(4.1) \quad \begin{cases} \frac{\partial u}{\partial t} - \delta(\|\nabla(K_{x,y} * u)\|) \nabla \cdot (\psi(\|\nabla u\|) \nabla u) + \alpha(u - u_0) = 0 \\ u(x, y, 0) = u_0(x, y), \quad \forall (x, y) \in \Omega, \\ u(t, x, y) = 0, \quad \forall (x, y) \in \partial\Omega \end{cases}$$

where $\alpha \in [0, 1)$, $\Omega \subseteq R^2$, $u_0 = Im * K_{x,y}$, Im is the original image, $K_{x,y}$ is a 2D filter kernel and

$$(4.2) \quad \psi : [0, \infty) \rightarrow [0, \infty), \quad \psi(s) = \varepsilon \left(\frac{\xi(u)}{|\beta s^2 + \zeta \ln \xi(u)|} \right)^{\frac{1}{3}},$$

where $\beta \in (2, 4]$, $\varepsilon, \zeta \in (0, 1)$ and $\xi(u) = v|\mu(\|\nabla u\|) + \text{median}(\|\nabla u\|)$, with $v \in [1, 3)$, $\mu(\cdot)$ returns the average and $\text{median}(\cdot)$ is median value of the argument. Also, the function $\delta : [0, \infty) \rightarrow [0, \infty)$, $\delta(s) = \kappa^{-r+1} \sqrt{v s^r + \rho}$, with $\kappa, r \in (0, 1)$, $v, \rho \in (1, 2)$.

The following explicit finite difference-based numerical approximation algorithm is constructed for it [4, 17]:

$$(4.3) \quad \begin{aligned} u_{i,j}^{n+1} &= u_{i,j}^n(1 - \alpha) + u_{i,j}^0\alpha + \delta_{i,j}(\psi_{i+\frac{1}{2},j}(u_{i+1,j}^n - u_{i,j}^n) \\ &\quad - \psi_{i-\frac{1}{2},j}(u_{i,j}^n - u_{i-1,j}^n) + \psi_{i,j+\frac{1}{2}}(u_{i,j+1}^n - u_{i,j}^n) - \psi_{i,j-\frac{1}{2}}(u_{i,j}^n - u_{i,j-1}^n)) \end{aligned}$$

A binary image is obtained by computing the absolute difference between 2 states, then a thresholding process [4]:

$$(4.4) \quad u_{i,j}^b = \begin{cases} 1, & \text{if } |u_{i,j}^{m+T} - u_{i,j}^m| \geq w \frac{\xi(u^{m+T}) + \xi(u^m)}{2}, \\ 0, & \text{otherwise} \end{cases},$$

$$\forall i \in \{1, \dots, I\}, \forall j \in \{1, \dots, J\}, w \in (0, 3).$$

Morphological operations are then applied to the binary image: dilation, thinning, gap filing, and small spots removal. The resulted image represents the edge detection and the pixels located in the vicinity of these extracted edges are then coded (4 or 8-neighborhood), A lossless RLE-inspired coding algorithm is applied to these sparse pixels. The sparse image is then transformed into a row vector R and at each step one determines the sequence $[R(i), n_i, z_i]$, where $(R(i) = R(i+1) = \dots = R(i+n_i-1))$ and z_i is the number of consecutive zeroes after the last occurrence. The current sequence is appended to a coding vector: $C = [C, R(i), n_i, z_i]$ [4].

In the image decompression stage, a decoding process is performed on the encoded pixels, first. Then, an interpolation is performed on the obtained sparsified image, by applying the following nonlinear compound fourth-order PDE-based structural inpainting model proposed by us in [7]:

$$(4.5) \quad \begin{cases} \frac{\partial u}{\partial t} + \lambda \delta(\|\nabla u\|) \nabla^2(\varphi(\|\Delta u\|) \nabla^2 u) - \eta \operatorname{div}(\psi(\|\nabla u\|) \nabla u) \\ u(x, y, 0) = u_0(x, y), \quad \forall (s, y) \in \Omega^+(1 - 1_\Gamma)(u - u_0) = 0 \\ \frac{\partial u}{\partial \vec{n}} = 0 \\ u(t, x, y) = 0, \quad \forall (x, y) \in \partial\Omega \end{cases}$$

where $\lambda \in [1.4, 2)$, $\eta \in (0.5, 1]$ and $\varphi : [0, \infty) \rightarrow [0, \infty)$, $\varphi(s) = \gamma \sqrt[k]{\frac{\xi(u)}{\zeta|\xi(u)+s|^{k+1}+\tau}}$, $\tau \in (2, 4]$, $\gamma, \zeta \in [0.5, 1)$, $k \in \{2, 5\}$. This PDE inpainting scheme is non-variational and combines nonlinear second and fourth order diffusions to achieve an effective interpolation. The next explicit iterative central difference-based numerical approximation algorithm is constructed for it [17]:

$$(4.6) \quad \begin{aligned} u_{i,j}^{n+1} &= u_{i,j}^n(1_\Gamma - \eta(\psi_{i+\frac{1}{2},j} + \psi_{i-\frac{1}{2},j} + \psi_{i,j+\frac{1}{2}} + \psi_{i,j-\frac{1}{2}})) \\ &\quad + \eta(u_{i+1,j}^n \psi_{i+\frac{1}{2},j} + u_{i-1,j}^n \psi_{i-\frac{1}{2},j} + u_{i,j+1}^n \psi_{i,j+\frac{1}{2}} + u_{i,j}^n \psi_{i,j-\frac{1}{2}}) \\ &\quad + u_{i,j}^0(1 - 1_\Gamma) - \lambda \delta_{i,j}(\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j}) \end{aligned}$$

for $n = 0, 1, \dots, N$, where N is large enough, depending on the number of sparse points [4].

The proposed PDE-based compression scheme has been tested successfully on hundreds of images and achieves good performance measure values: compression rate, compression ratio, fidelity and quality. It outperforms BTTC-L, the most PDE-based methods, is slightly better than JPEG at high compression rates, but it is outperformed by the JPEG 2000 and R-EED codecs. The main steps of this compression process are described in Fig. 8 [4].

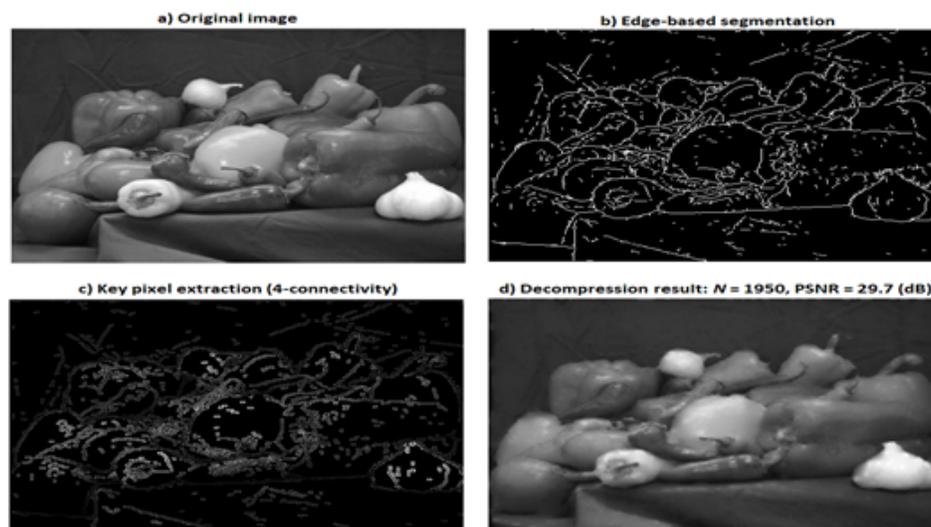


Figure 8: Stages of a compression process, at ratio = 12:1

Method comparison results, representing PSNR and SSIM values achieved by state of the art PDE compression models, are displayed in Table 2.

Table 2. Method comparison: PSNR and SSIM at compression rate 0.4 bpp

Compression technique	PSNR	SSIM
The proposed AD-based technique	27.8777 (dB)	0.8231
Linear homogeneous diffusion	20.8971 (dB)	0.6866
Biharmonic smoothing operator	21.4235 (dB)	0.7230
BTTC-L compression scheme	22.8565 (dB)	0.7487
BTTC-EED compression	25.6607 (dB)	0.7738
R-EED codec	28.9994 (dB)	0.8586
JPEG	27.7475	0.8143
JPEG 2000	28.4545 (dB)	0.8467

5 Conclusions

A survey of the state of the art PDE-based image compression techniques has been presented in this work. These approaches use the partial differential equations mostly for the decompression task and use other coding schemes for compression.

The PDEs could also be useful in the pre-processing stages of the compression that are related to image enhancement: image filtering, restoration, inpainting. Most PDE-based image compression models have a variational character, being derived from some energy cost functional minimization schemes.

The PDE-based image compression may represent a better alternative to transform-based codecs, such as JPEG that is based on Discrete Cosinus Transform (DCT) and JPEG 2000 that uses the Discrete Wavelet Transform. While the transform-based JPEG compression is seriously affected by block artifacts, since discrete cosine transform is computed within blocks of $[8 \times 8]$, the PDE-based codecs do not have this shortcoming.

Our own contribution to the PDE-based compression domain has been also described here. Unlike other techniques in this field, the proposed nonlinear diffusion-based compression framework uses PDE models in both compression and decompression stages. Also, unlike the PDE-based schemes used by many other compression approaches, our nonlinear anisotropic diffusion models are novel and non-variational. Unfortunately, it cannot be successfully used for textured image compression, given its structural interpolation model used in the decompression stage.

References

- [1] G. Aronsson, *Extension of functions satisfying Lipschitz conditions*, Arkiv för Matematik, 6 (28) (1967), 551-561 (cited on pp. 38, 61).
- [2] G. Aubert, P. Kornprobst, *Mathematical problems in image processing: partial differential equations and the calculus of variations*, Vol. 147, Springer Science & Business Media, 2006.
- [3] T. Barbu, *Robust contour tracking model using a variational level-set algorithm*, Numerical Functional Analysis and Optimization 35 (3) (2014), 263-274.
- [4] T. Barbu, *Segmentation-based non-texture image compression framework using anisotropic diffusion models*, Proceedings of the Romanian Academy, Series A: Mathematics, Physics, Technical Sciences, Information Science 20 (2) (2019), 122-130.
- [5] T. Barbu, *Variational image inpainting technique based on nonlinear second-order diffusions*, Computers & Electrical Engineering, 54 (2016), 345-353.
- [6] T. Barbu, A. Favini, *Rigorous mathematical investigation of a nonlinear anisotropic diffusion-based image restoration model*, Electronic Journal of Differential Equations 2014 (129) (2014), 1-9.
- [7] V. Bhaskaran, K. Konstantinides, *Image and Video Compression Standards*, Kluwer Academic Press, Boston, 1995.
- [8] T.F. Chan and J. Shen, *Morphologically invariant PDE inpaintings*, UCLA CAM Report, 2001, 1-15.
- [9] J. Chen, F. Ye, J. Di, C. Liu, A. Men, *Depth map compression via edge-based inpainting*, Proceedings of the 29th Picture Coding Symposium, 57-60, Kraków, Poland, May 2012.
- [10] L. Demaret, N. Dyn, A. Iske, *Image compression by linear splines over adaptive triangulations*, Technical report, Dept. of Mathematics, University of Leicester, UK, January 2005.

- [11] R. Distasi, M. Nappi, S. Vitulano, *Image compression by B-tree triangular coding*, IEEE Transactions on Communications 45 (9) (1997), 1095-1100.
- [12] I. Galic, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, H.-P. Seidel, *Image compression with anisotropic diffusion*, Journal of Mathematical Imaging and Vision 31 (2-3) (2008), 255-269.
- [13] I. Galic, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, H.-P. Seidel, *Towards PDE-based image compression. Variational, Geometric and Level-Set Methods in Computer Vision*, Lecture Notes in Computer Science, vol. 3752, Springer, Berlin, 2005, 37-48.
- [14] R. Gonzalez and R. Woods, *Digital Image Processing*, Prentice Hall, New York, NY, USA, 2nd ed., 2001.
- [15] L. Hoeltgen, P. Peter, M. Breuß, *Clustering-based quantization for PDE-based image compression*, Signal, Image and Video Processing 12 (3) (2018), 411-419.
- [16] S. Hoffmann, G. Plonka, and J. Weickert, *Discrete Green's functions for harmonic and biharmonic inpainting with sparse atoms*, International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, 2015, 169-182.
- [17] P. Johnson, *Finite Difference for PDEs*, School of Mathematics, University of Manchester, Semester I, 2008.
- [18] S. Lee, E. Park, *A content adaptive fast PDE algorithm for motion estimation based on matching error prediction*, Journal of Communications and Networks 12.1 (2010), 5-10.
- [19] M. Mainberger, J. Weickert, *Edge-based image compression with homogeneous diffusion*, Computer Analysis of Images and Patterns, Lecture Notes in Computer Science, Vol. 5702, Springer, Berlin, 2009; 476-483.
- [20] D. Marr, E. Hildreth, *Theory of edge detection*, Proceedings of the Royal Society of London. Series B, Biological Sciences, 207 (1167) (1980), 187-217.
- [21] D. Mumford, J. Shah, *Optimal approximation by piecewise smooth functions and associated variational problems*, Comm. Pure Appl. Math. 42 (1989), 577-685.
- [22] P. Peter, J. Weickert, *Colour image compression with anisotropic diffusion*, Proc. of 21st IEEE International Conference on Image Processing, Paris, France, 2014, 4822-4826.
- [23] P.T. Peter, *Understanding and advancing PDE-based image compression*, PhD Thesis, Saarland University, 2016.
- [24] K. Sayood, *Introduction to Data Compression*, Morgan Kaufmann, Third edition, 2005.
- [25] C. Schmaltz, J. Weickert, A. Bruhn, *Beating the quality of JPEG 2000 with anisotropic diffusion*, Pattern Recognition, Lecture Notes in Computer Science Vol. 5748, Springer, Berlin, 2009, 452-461.
- [26] C. Schmaltz, P. Peter, M. Mainberger, F. Ebel, J. Weickert, A., Bruhn, *Understanding, optimizing, and extending data compression with anisotropic diffusion*, International Journal of Computer Vision, 108 (3) (2014), 222-240.
- [27] C.B. Schonlieb, *Partial Differential Equation Methods for Image Inpainting*, Vol. 29, Cambridge University Press, 2015.
- [28] J. Weickert, *Anisotropic Diffusion in Image Processing*, European Consortium for Mathematics in Industry. B.G. Teubner, Stuttgart, Germany, 1998.

- [29] J. Zhang, K. Chen, *Variational image registration by a total fractional-order variation model*, Journal of Computational Physics 293 (2015), 442-461.

Author's address:

Tudor Barbu
Institute of Computer Science of the Romanian Academy - Iași Branch,
Iași, Romania.
E-mail: tudor.barbu@iit.academiaromana-is.ro