

RESOLUTION ENHANCEMENT OF COLORED IMAGES BY INVERSE DIFFUSION PROCESSES

N. Sochen *Y. Y. Zeevi*

Department of Electrical Engineering
Technion-Israel Institute of Technology
Haifa 32000, Israel.

ABSTRACT

Algorithms for resolution enhancement are needed in various applications of image processing and communication such as compression, and HDTV.

We develop a geometrical algorithm, based on diffusion processes which are used both for smoothing of colored images and enhancement of the colored images. The latter is accomplished by “solving” an inverse diffusion problem which is ill posed. In order to stabilize the flow, as well as to enhance important features (e.g. edges) on the expense of less important image domains, we use a modified Beltrami diffusion equation. Results indicate that it is possible to impose the backwards flow, in spite of the instabilities, and thereby enhance the image up to a certain level of resolution which depends on the nature of the image.

1. INTRODUCTION

In recent years there has been a great deal of interest in various aspects of image enhancement. Since the bottle neck of various communication channels does not permit transmission of high resolution colored images with currently existing compression techniques, some extra processing in the form of image enhancement (“super-resolution”) is necessary [8].

We develop a geometrical approach where diffusion processes are used both for smoothing the enlarged image, when “time” is flowing forward, and for enhancement. The latter is accomplished by allowing the “time” to flow backwards, i.e. “solving” an inverse diffusion process which is mathematically ill posed. In order to stabilize the flow, as well as to enhance important features (e.g. edges) on the expense of less important image domains, we modify the diffusion equation.

Let us first look at a one-dimensional simple example in order to simplify the problem. Suppose we have a one-dimensional lattice \mathcal{L} , of order $|\mathcal{L}| = M$. The one-dimensional “image” is given by the amplitude at the lattice points, that is, I_i for $i \in \mathcal{L}$. We now want to

zoom-in by a super-resolution algorithm, and to obtain more pixels with an appropriate higher resolution (i.e. more detail). To this end, we multiply the number of sampling points by m and construct a larger lattice \mathcal{L}^s . We then embed \mathcal{L} in \mathcal{L}^s in a natural way. The question now is what should be the new values I_i^s for $i \in \mathcal{L}^s$? We call I^s the “zooming-in” (or super-resolution) of I . It is shown elsewhere [7] that the following relation should be satisfied:

$$(I_x)_i = (I_x^s)_{mi} \quad i = 1, 2, \dots, M, \quad (1)$$

where the subscript x means derivative in the x direction. For higher dimensional cases we equate each one of the partial derivatives.

Since numerical derivatives are approximated by differences in our computational implementations, this condition requires a wider dynamical range of I^s . In fact, the required dynamical range may be wider than the bounds that image display permit. Obviously, the problem is compounded in the higher dimensional color space.

We, therefore, have to resort to non-linear rescaling which preserves and enhances sharp transitions in intensity on the expense of the lower gradients which are less important in vision.

In order to treat large gradients (i.e. edges) differently from more homogeneous regions, we use the Beltrami operator [6] which after some modifications becomes most suitable for our enhancement needs [3]. We also add geometrical constraints to prevent the intensity from blowing up.

2. EMBEDDING AND ENLARGEMENT

We take the original image as a two-dimensional lattice \mathcal{L} of size $N \times M$, where the intensity at each pixel is given by I_{ij}^a , where $a = r, g, b$. We then multiply the x coordinate by a factor m and the y coordinate by n , to get a $mM \times nN$ lattice \mathcal{L}^s . (Note that in most image processing applications it is required that $m = n$.)

The problem can now be formulated as follows: How should one fix the values of the intensity I^s at the pixels such that the enlarged image on \mathcal{I}^s looks like a finer scale image (i.e. enhanced as to appear sharper and with greater detail) of the same scenario, given only the lower scale which is the image on the lattice \mathcal{I} .

We proceed in three steps:

1) Embedding: Define the auxiliary sub-lattice \mathcal{I}_e as follows:

$$\mathcal{I}_e = \{(i, j) \in \mathcal{I}^s \mid i \bmod m = j \bmod n = 0\}, \quad (2)$$

where m and n are small integers for each step of the lattice expansion. Then

$$(I^s)_{ij}^a = \begin{cases} I_{i/m, j/n}^a & \text{if } (i, j) \in \mathcal{I}_e \\ I_{[i/m], [j/n]}^a + \Omega & \text{if } (i, j) \in \bar{\mathcal{I}}_e \end{cases} \quad (3)$$

where $\bar{\mathcal{I}}_e$ is the compliment of \mathcal{I}_e in \mathcal{I}^s , and Ω is i.i.d. random variable with a small variance compared with the range of the given intensities, and $[x]$ denotes the integer part of x .

2) Enlargement: We interpolate all pixels that do not belong to \mathcal{I}_e by solving a diffusion process, given the sub-lattice \mathcal{I}_e as a boundary condition, namely:

$$(I_t^s)_{ij}^a = (\Delta I^s)_{ij}^a = (I_{xx}^s)_{ij}^a + (I_{yy}^s)_{ij}^a, \quad \forall (i, j) \in \bar{\mathcal{I}}_e. \quad (4)$$

In other words, we interpolate by means of a potential surface in a way that conserves smoothness. One can also try to interpolate by a minimal surface (with the Beltrami operator for example) but this is dangerous because the minimal surface, depending on the boundary conditions, may break and change topology. It is, thus, safer to stick with the potential surface unless a good a-priori knowledge of the image suggests otherwise.

3) Enhancement: Various possibilities are described in the next two sections.

3. ENHANCEMENT BY INVERSE BELTRAMI FLOW

The Beltrami operator is the natural generalization of the Laplacian from the plane to geometrically non trivial surfaces (or higher dimensional manifolds).

It is expressed by means of the local metric on the surface which we take as the induced metric. Explicitly, for grey level images, given as a function $I(x, y)$, we think of the image as a two-dimensional surface embedded in a three-dimensional Euclidean space as follows:

$$(x, y) \rightarrow (x, y, I(x, y)). \quad (5)$$

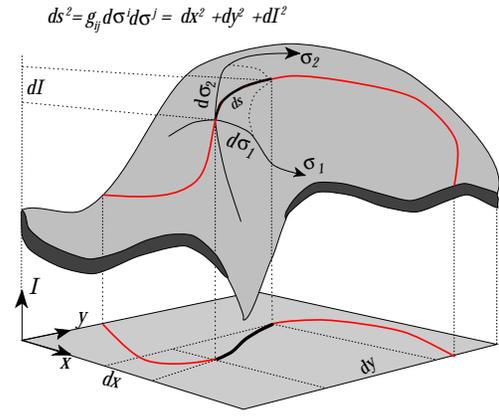


Figure 1: Line element form two view points: intrinsic and extrinsic.

The induced metric is than the following symmetric and positive definite matrix:

$$G = \begin{pmatrix} 1 + I_x^2 & I_x I_y \\ I_x I_y & 1 + I_y^2 \end{pmatrix}. \quad (6)$$

We denote the elements of the metric by $g_{\mu\nu}$, those of the inverse matrix by $g^{\mu\nu}$, and denote by g the determinant of the metric. The form of the metric, Eq. (6), can be understood in a simple way. It expresses the fact that the length of infinitesimal curve segment on the image surface can be measured on the surface or in the embedding space with the same result (see Fig. 1):

$$\begin{aligned} ds^2 &= g_{\mu\nu} d\sigma^\mu d\sigma^\nu = dx^2 + dy^2 + dI^2 \\ &= dx^2 + dy^2 + (I_x dx + I_y dy)^2 \\ &= (1 + I_x^2) dx^2 + 2I_x I_y dx dy + (1 + I_y^2) dy^2, \end{aligned} \quad (7)$$

where we identify the coordinates on the image surface (i.e. (σ^1, σ^2)) with (x, y) , and use Einstein summation convention (i.e. indices that repeat twice are summed over).

Similarly, we treat color images as an embedding of a two-dimensional surface in a five-dimensional space

$$(x, y) \rightarrow (x, y, R(x, y), G(x, y), B(x, y)), \quad (8)$$

and the induced metric is

$$G = \begin{pmatrix} 1 + R_x^2 + G_x^2 + B_x^2 & R_x R_y + G_x G_y + B_x B_y \\ R_x R_y + G_x G_y + B_x B_y & 1 + R_y^2 + G_y^2 + B_y^2 \end{pmatrix}. \quad (9)$$

The Beltrami flow is than written as a gradient descent equation

$$I_t^a = -\frac{1}{\sqrt{g}} h^{ab} \frac{\delta S}{\delta I^b}, \quad (10)$$

where the action functional S is given by

$$S[X^i, g_{\mu\nu}, h_{ij}] = \int d^2\sigma \sqrt{g} g^{\mu\nu} \partial_\mu X^i \partial_\nu X^j h_{ij}(\mathbf{X}). \quad (11)$$

For color images $(X^1, X^2, X^3, X^4, X^5) = (x, y, R, G, B)$, and the metric of the embedding space is h_{ij} . By standard methods of calculus of variations we derive the Beltrami diffusion flow (see [5] for derivation)

$$I_t^a = \Delta_g I^a = \frac{1}{\sqrt{g}} \partial_\mu (\sqrt{g} g^{\mu\nu} \partial_\nu I^a), \quad (12)$$

which can be written in a vector-matricial form:

$$I_t^a = \Delta_g I^a = \frac{1}{\sqrt{g}} \text{Div}(\sqrt{g} G^{-1} \nabla I^a). \quad (13)$$

The Beltrami operator $\Delta_g I$ is an adaptive smoothing operator that does not affect edges as much as it smooths the homogeneous regions of the image. Taking the inverse of the Beltrami operator, $1/\Delta_g I$, enhances the edges more than it affects other regions. Since inverse Beltrami is a highly singular operation, we regularize it by means of an exponential function,

$$\text{Sign}(\Delta_g I^a) e^{-\lambda_1 |\Delta_g I^a|}, \quad (14)$$

with a decay factor λ_1 as a free parameter of the regularization (selected in accordance with the instability encountered in the implementation).

Obviously this is a highly non-stable flow and one should proceed with care. In order to stabilize it, we extract from the original image the metric $(g_{\mu\nu})$ that describes the local geometry of the image. The determinant of the metric, g , is a good measure for the gradient at the point. In the extended image, the gradients at the original image sub-lattice are smaller than the gradients of the original image, simply because the same change at the value of the pixels is now smeared over a larger distance. We would like to stretch the image in the intensity direction such that the metric of the enlarged and stretched image $(g_{\mu\nu}^s)$ at the original image-sub-lattice, e will be as close as possible to the values of the original image. We add to this effect a second term which controls the amount of change in the intensity values according to the distance between the zooming-in metric and the original image:

$$e^{-\lambda_2 (g^s - g)} (I^{s^a} - I^a) \delta((x, y) \in e), \quad (15)$$

where, again, λ_2 is a parameter to be fine tuned by the user. This condition is in the spirit of Eq. (1).

4. ENHANCEMENT BY A MODIFIED INVERSE BELTRAMI FLOW

Since the metric is a symmetric, positive definite, bilinear form, we can diagonalize it as follows:

$$G = U^T \Lambda U, \quad (16)$$

where U is an orthogonal matrix that is built from the eigenvectors of G . In particular for our grey level image's metric,

$U_1^T = (I_x, I_y) / \sqrt{I_x^2 + I_y^2}$ is the gradient direction, and $U_2^T = (-I_y, I_x) / \sqrt{I_x^2 + I_y^2}$ is the perpendicular direction.

To enhance edges, we modify the metric as follows:

$$\tilde{G} = U^T \begin{pmatrix} -1/\alpha & 0 \\ 0 & \alpha \end{pmatrix} U, \quad (17)$$

where $-1/\alpha$ replaces the larger eigenvalue and α replaces the second eigenvalue. Substituting this new "metric" in the Laplace-Beltrami operator, and calculating all the derivatives explicitly, we get the unstable inverse heat equation:

$$I_t = -\alpha (I_{xx} + I_{yy}). \quad (18)$$

In order to stabilize the flow we first smooth a copy of the image by a convolution with a Gaussian (or equivalently solving numerically the heat equation), and then extract the metric from the smoothed image. In this way we are guaranteed that the metric has all the nice properties, since it is the metric of an image. Denote by \hat{I}_ρ^a the smoothed color image components:

$$\hat{I}_\rho^a = \int dx_0 dy_0 \frac{1}{4\pi\rho} e^{-\frac{(x-x_0)^2 + (y-y_0)^2}{4\rho}} I^a(x_0, y_0), \quad (19)$$

then the smoothed metric is ¹:

$$\hat{G} = \hat{U}^T \hat{\Lambda} \hat{U}. \quad (20)$$

Note also that $|\hat{g}| = 1$. The modified flow now reads

$$\begin{aligned} I_t^a &= \frac{1}{\sqrt{|\hat{g}|}} \text{Div}(\sqrt{|\hat{g}|} \hat{G}^{-1} \nabla I^a) \\ &= \text{Div}(\hat{G}^{-1} \nabla I^a) = \text{Div}(\hat{U}^T \begin{pmatrix} -\alpha & 0 \\ 0 & 1/\alpha \end{pmatrix} \hat{U} \nabla I^a). \end{aligned} \quad (21)$$

We may, as before, add local constraints to prevent the intensity from blowing up.

The idea of stabilizing the inverse heat equation is not new in image processing. Few references among a vast literature are the "shock filters" [4] for grey-level, and its extension [1].

¹This smoothing is similar to [2], and slightly different from [3] where the modified Beltrami flow were first used.

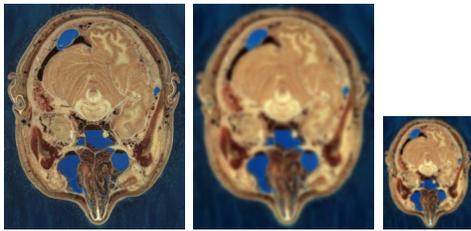


Figure 2: (a): Original, (b): After linear smoothing, (c): After size reduction.

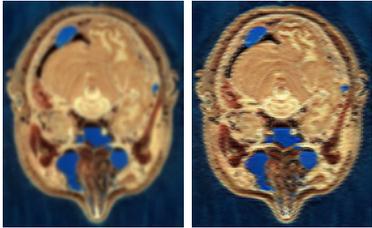


Figure 3: (a): Enlargement of Fig. (2c). (b): Enhanced version of the enlarged image (a).

5. RESULTS

We choose a colored MRI section of the brain, Fig. (2a), and diffuse it linearly to get the image depicted in Fig. (2b). We then reduce it by half to get our source image, Fig. (2c). Our algorithm starts with the image of Fig. (2c). After the enlargement and smoothing stage, the obtained image is depicted at Fig. (3b). The enhancement result is shown in Fig. (3b). Note that processing is executed in the five-dimensional space of the colored images. We display only the achromatic component of the original and computed images.

6. SUMMARY AND CONCLUSIONS

We have proposed a zooming-in condition Eq. (1) which is valid in a situation where the dynamical range is unbounded or large enough for the condition to have a solution. For the generic case, this condition is too stringent and we have to weaken the condition in an intelligent way. We opt for a distinction between higher and lower gradients or, for the two-dimensional case, for a distinction between higher values of the determinant of the metric and lower values of it. In the case of the higher values we tried to keep this condition while leaving the lower values to depart greatly from it. The image surface is trying, at the same time, to conserve its smoothness as much as possible under the imposed conditions.

Some results that show the effects and efficacy of

this algorithm are shown in Sec. 5. One should keep in mind that this is only a preliminary study and various aspects should be tackled. Numerical derivatives may be refined to get a smoother result, color space geometry and coordinates should be studied for a better representation and results. Finally, an objective functional for quality measurement should be constructed to get an optimal fine tuning of the various parameters of the algorithm. The latter is of course a general problem in assessment of image quality. As a by product our study may result in some guidelines for the development of such an image quality functional.

Acknowledgment:

We would like to thank Ron Kimmel for useful discussions. This research has been supported in part by the Ollendorff Center, the Fund for Promotion of Research at the Technion, and by the Ministry of Science Vision Platform Project.

7. REFERENCES

- [1] L. Alvarez and L. Mazora. Signal and image restoration using shock filters and anisotropic diffusion. *SIAM J. Numer. Anal.*, 31:590–605, 1994.
- [2] G. H. Cottet and L. Germain. Image processing through reaction combined with nonlinear diffusion. *Math. Comp.*, 61:659–673, 1993.
- [3] R. Kimmel R. Malladi and N. Sochen. Images as embedded maps and minimal surfaces: Movies, color, texture, and volumetric medical images. LBNL ,UC and TAUHEP Report LBNL-40490, UC-405, June 1997.
- [4] S.J. Osher and L.I. Rudin. Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.*, 27(4):919–940, August 1990.
- [5] N. Sochen, R. Kimmel, and R. Malladi. From high energy physics to low level vision. Report LBNL 39243, LBNL, UC Berkeley, CA 94720, August 1996.
- [6] N. Sochen R. Kimmel and R. Malladi. A general framework for low level vision. *IEEE Trans. IP*, 1997, in press.
- [7] N. Sochen and Y.Y. Zeevi. Resolution enhancement of colored images by inverse non-linear scale-space. *in preparation*, 1997.
- [8] I. Vitsnudel, Y.Y. Zeevi and R. Ginosar. Neural network aided design for image processing. *Proc. SPIE*, 1606:1089–1091, 1991.