1. Introduction

The concept of image editing encompasses the operations by which the local content of one or several images is selected and manipulated to create new syncretic images. The simplest such operation is a copy-paste of a part of an image into another. Image stitching, by which several images are fused into a panorama is another example (Levin et al., 2004). Local contrast or color adjustments after selecting manually a part of the image (for example the shadows) is another variant. The inpainting operation, by which an object is removed in the image and replaced by a texture (Chuan et al., 2008; Jia et al., 2006). As amply demonstrated by Pérez et al. (2003), the Poisson equation (1) is an extremely efficient response to this problem:

\[
\Delta u(x, y) = \int f(x, y) \, dx + \int g(x, y) \, dy
\]

where \( \Delta u(x, y) = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \) is the Laplacian of \( u \). The authors of Pérez et al. (2003) proposed to use (1) for “seamless editing”. They rely on psychophysical observations suggesting that our vision perceives the Laplacian of the images rather of the image itself. Thus the image \( u \) is satisfactorily specified by giving its Laplacian on the boundary is enough to reconstruct the image, and this is done by the Poisson equation. The main challenge in image editing is to avoid suspicious color or texture alterations which would reveal the silhouette \( \partial \Omega \) of the image region which has been modified (Chuan et al., 2008; Jia et al., 2006). As amply demonstrated by Pérez et al. (2003), the Poisson equation (1) is an extremely efficient response to this problem.
accelerate the convergence of basic iterative methods like SOR. Multigrid methods can treat arbitrary regions and boundary conditions, but are notoriously slow when the regions have not a simple geometry. Indeed, they proceed by a recursive subsampling of the domain \( \Omega \). This is only efficient when \( \Omega \) is a rectangle with dyadic dimensions (or the union of a few rectangles with dyadic dimensions). Efficient solvers comparable to the multigrid method to solve the Poisson editing equations are examined in (Jeschke et al., 2009) where the authors present a Laplacian solver for minimal surfaces. In (Agarwala, 2007) the author proposes to use quad-trees to improve the efficiency of the technique that constructs seamless composites. Later solutions have also relied on the multigrid method: (Kazhdan and Hoppe, 2008; McCann and Pollard, 2008; Grady et al., 2008).

Our goal here is to show that a simple global reformulation of the Poisson editing problem permits to use fast Fourier solvers. A Fourier solver for the Poisson equation is exact, non-iterative and in practice faster than a full multigrid method, even in the case of a rectangular domain.

Since the proposed implementation method solves the Poisson equation over the whole image domain instead of only the unknown region \( \Omega \), it will be verified experimentally that the known pixel values are only minimally altered.

The proposed numerical method permits to automatize a local contrast enhancement proposed in (Pérez et al., 2003). The new tool is fast and efficient, requiring no manual selection and allowing one to edit regions with arbitrarily complex topology with no additional cost.

A review of Poisson image editing is presented in Section 2. Section 3 details the new implementation using the Fourier method. A number of possible choices for the guidance vector field is examined. These different choices with experimental results are presented in Section 4. Finally, Section 5 shows some examples of the correctness of the method.

2. Poisson image editing

In this Section, we detail the image Poisson reconstruction from “a guidance vector field”, in the terminology of Pérez et al. (2003). Let \( R \), a closed subset of \( \mathbb{R}^2 \), be the image domain, and let \( \Omega \subset R \) be a closed subset with boundary \( \partial \Omega \). The problem is to find the image whose gradient field is the closest, in \( L^2 \)-norm, to the prescribed “guidance vector field” \( \mathbf{v} \) defined in \( \Omega \), under given boundary conditions on \( \partial \Omega \). The image is known in the rest of the image, \( R \setminus \Omega \) (Fig. 1). This problem writes

\[
\min_u \int_\Omega |\nabla u - \mathbf{v}|^2, \quad \text{with} \quad u|_{\partial \Omega} = f|_{\partial \Omega}
\]  

and its solution is the unique solution of the Poisson equation with Dirichlet boundary conditions

\[
\Delta u = \nabla \cdot \mathbf{v}, \quad \text{with} \quad u|_{\partial \Omega} = f|_{\partial \Omega}.
\]

The variational problem (2) can be discretized using a finite difference approach. Using the notation of Pérez et al. (2003), it amounts to solve the following linear equations

\[
|N_p| u_p - \sum_{q \in N_p \setminus \Omega} u_q = \sum_{q \in N_p \setminus \Omega} f_q + \sum_{q \in N_p} \nu_v, \quad \text{for all} \quad p \in \Omega,
\]  

where \( N_p \) is the 4-connected neighborhood of \( p \) and \( \nu_v = \mathbf{v}(\frac{p + q}{2}) \cdot \overrightarrow{pq} \). Since the region \( \Omega \) can have an arbitrary shape, the authors in (Pérez et al., 2003) propose to solve the linear system (4) using iterative solvers, for example the Gauss–Seidel iteration with overrelaxation or a multigrid method.

3. How to solve the Poisson image editing equations using Fourier transform?

The aim of this work is to solve the above variational problem (2), and, consequently the associated PDE (3), using the Fourier method. To do that the boundary of the domain \( \Omega \) must coincide with the coordinate lines, which in general is not the case. This can be done by extending the definition domain of the guidance vector field to the whole image domain \( R \), and solving the variational problem over all the whole image domain. The guidance vector field \( \mathbf{v} \) defined on \( \Omega \) is extended to \( R \) by (Fig. 2)

\[
\mathbf{v} = \begin{cases} \mathbf{v} & \text{over } \Omega \\ \nabla f & \text{otherwise}. \end{cases}
\]

where \( f \) is the image, which is known over \( R \setminus \Omega \). Then the problem is to minimize

\[
\min_u \int_R |\nabla u - \mathbf{v}|^2.
\]

The minimizer is uniquely determined by the Euler–Lagrange equation

\[
\Delta u = \nabla \cdot \mathbf{v}, \quad \text{over} \quad R,
\]

with homogeneous Neumann boundary conditions

\[
\frac{\partial u}{\partial n} = 0 \quad \text{over} \quad \partial R.
\]

where \( n \) is the direction orthogonal to the boundary.

By the above result the solution of the minimization problem (6) is the restriction of an even function which is periodic with a period twice the image size. The gradient field of the solution is the vector field that best approximates, in the \( L^2 \)-norm, the guidance vector field \( \mathbf{v} \). In \( R \setminus \Omega \) the gradient function approximates the gradient of \( f \) and in that sense the solution is only an approximation of \( f \). Yet, their Laplacians coincide, which means that they differ only by a very smooth (harmonic) function. In \( \Omega \) the gradient of the solution approximates the given guidance vector field defined in \( \Omega \).
3.1. Fourier transform method

Since the Eq. (7) has constant coefficients and the boundaries of $R$ coincide with the coordinate lines, we can apply the Fourier transform method to solve the partial differential equation. The Neumann boundary condition is implicitly imposed by extending the original image symmetrically across its sides, so that the extended image, which is four times bigger, becomes symmetric and periodic. Once this is done, we can apply the Fourier transform.

The discrete Fourier transform (DFT) permits to compute directly the Fourier coefficients of a band limited and periodic function $u$ from its samples $u_{ij}$ on a $J \times L$ grid by

$$
\hat{u}_{mn} = \sum_{i=0}^{J-1} \sum_{j=0}^{L-1} u_{ij} e^{-2\pi i m i/J} e^{-2\pi i n j/L},
$$

(9)

with $m = 0, \ldots, J - 1$ and $n = 0, \ldots, L - 1$. The Fourier series formula recovers $u$ from its Fourier coefficients by

$$
u(x,y) = \frac{1}{J L} \sum_{m=0}^{J-1} \sum_{n=0}^{L-1} \hat{u}_{mn} e^{2\pi i m x/J} e^{2\pi i n y/L}.
$$

(10)

Thus, by a simple differentiation, the Poisson equation $\Delta u = \text{div}(V)$ translates into a relationship between the Fourier coefficients of $u$ and $V$:

$$
\left(\frac{4\pi m}{J} \right)^{2} + \left(\frac{4\pi n}{L} \right)^{2} \hat{u}_{mn} = \frac{2\pi i m}{J} \hat{V}_{1mn} + \frac{2\pi i n}{L} \hat{V}_{2mn},
$$

(11)

where $V = (V_1, V_2)$.

In short, the strategy for solving (7–8) by Fourier technique is:

- Quadruplicate by symmetry the discrete domain and $V$.
- Compute the discrete Fourier transforms of $V_1$ and $V_2$.
- Compute the discrete Fourier transform of the solution $\hat{u}_{mn}$ as

$$
\hat{u}_{mn} = \left(\frac{4\pi m}{J} \right)^{2} + \left(\frac{4\pi n}{L} \right)^{2} \hat{V}_{1mn} + \left(\frac{4\pi m}{J} \right) \hat{V}_{2mn}.
$$

(12)

- Obtain the samples $u_{ij}$ of the solution by the inverse discrete Fourier transform.
- Restrict them to the initial domain.

Remark. The Fourier transform comes with some overhead such as padding to the nearest power of 2, but with smart Fourier libraries, like fftw, having integer values that are not powers of 2 is no more a complexity issue. Products of small factors are most valuable, and it is anyway easy to change them interactively: the level $j$ such that the level set $\{x | u(x) < j\}$ has its area equal to 25% of the image area. Then $\alpha = 2.5$, as indicated. With these default values for $T$, $\alpha$ the proposed algorithm becomes fully automatic, and it is anyway easy to change them interactively:

- Select the dark region $\Omega$ by the threshold $T$.
- Define the guidance vector field by

$$
V = \begin{cases} 
\nabla f & \text{in } R \setminus \Omega, \\
\alpha \nabla f & \text{in } \Omega,
\end{cases}
$$

(15)

where $f$ is the gray level intensity.

4. Experimental results

Now we list some of the possible choices for the guidance vector field and show the results of the proposed implementation. In some examples a classic multigrid method would work as well. Yet, for all examples where the structure of $\Omega$ is complicated, using the global Fourier strategy simplifies drastically the situation.

4.1. Local illumination changes

Perez et al. use the method of Fattal et al. (2002) to modify smoothly the image dynamic range. The idea is to transform the gradient field of the logarithm of the image to reduce the large gradients and to increase the small ones. The transformed gradient is used to reconstruct the logarithm of the image, by solving the Poisson equation. In fact in (Pérez et al., 2003) the authors select a region $\Omega$, for example an under-exposed region, and transform the gradient in the log-domain by

$$
\nu = \alpha \left| \nabla f \right|^{-\beta} \nabla f,
$$

(14)

where $\alpha = 0.2$ times the average gradient norm of $\Omega$ and $\beta = 0.2$, that is, the guidance vector field is a concave function of the gradient in the log-domain.

The Perez et al. method is relatively complex. Indeed, it depends on two parameters, $\alpha, \beta$. The selection of the dark region is manual and only permits to select a few simple image regions. Thus, it is not adapted to treat the dark regions of images with back light or excessively dark shadows. Indeed, in such images the dark regions can be many and have a complicated topology. Thus, they can hardly be selected manually. The alternative proposed here is to select the dark region using a threshold $T$. $T = 50$ was the default threshold used successfully in all treated examples. Applying this threshold in classic JPEG images to the gray level intensity yields automatically the perceptually dark regions. Over the dark region, the gradient vector of the gray level intensity is amplified by a factor $\alpha$. Since a stronger factor would amplify too much the noise, the default value was 2.5.

These values ($T = 50, \alpha = 2.5$) are easily explained. First of all most eight bit digital images have their parts with luminance below 50 quite dark and no detail can be really seen on most screens below this value. It is therefore natural to try to enhance contrast in these regions. If such dark regions are small then the method needs not be applied, no image improvement is possible. If instead the dark regions are large, this contrast enhancement can add up valuable visual information. Now the question is: why $\alpha = 2.5$ as default value? Here again, we assume a decent quality image, where the noise standard deviation does not exceed four. Then multiplying the contrast by 2.5 gives back a final noise of standard deviation less than 10, which is the limit where the noise is still tolerable for users.

In short, the default values $T = 50$ and $\alpha = 2.5$ are not universal. They are motivated by the average quality eight bit images and the average sensitivity of visual perception in dark regions. If the method has to become automatic, then the best method is to select the level $\lambda$ such that the level set $\{x | u(x) < \lambda\}$ has its area equal to 25% of the image area. Then $\alpha = 2.5$, as indicated. With these default values for $T, \alpha$ the proposed algorithm becomes fully automatic, and it is anyway easy to change them interactively.
Solve the Poisson equation with Neumann boundary conditions using the Fourier transform as explained in the previous section.

Compute the color channels proportionally using the new gray level intensity.

Figs. 3–5 show results obtained using the proposed algorithm. Note the complexity of the topology dark regions. The method can be tried online on any uploaded image at http://www.ipol.im/pub/demo/imps_selective_contrast_adjustment/.

4.2. Other editing operations

The goal of this short section is to illustrate the fact that the proposed Fourier method gives equivalent results to multigrid methods. In the classic copy-paste problem, the guidance vector field $v$ is a gradient field from another source image $g$. The guidance field therefore is

$$V = \begin{cases} \nabla g, & \text{over } \Omega, \\ \nabla f, & \text{otherwise,} \end{cases}$$

where $f$ be the image which will be kept on $\mathbb{R} \setminus \Omega$. Fig. 6 shows the source image $g$, the original image $f$ and the obtained result.

The same technique also can be used as inpainting method, using as source image the destination image itself, typically to remove an object and to replace it by a texture patch from the same image. In Fig. 7 we can observe an example of this application.

Another application proposed in Perez et al. involves the guidance vector field

Fig. 3. Top left: original image. Top right: zones with intensity below $T = 50$. Bottom: result using guidance vector field (15) with $\alpha = 2.5$.

Fig. 4. Top left: original image. Top right: zones with intensity below $T = 50$. Bottom: result using guidance vector field (15) with $\alpha = 2.5$. 
\[ V = \begin{cases} \nabla f, & \text{if } |\nabla f| > |\nabla g|, \\ \nabla g, & \text{otherwise}. \end{cases} \]  

(17)

In this case, two images are mixed into one, thus creating a realistic graffiti (Fig. 8).

5. Correctness of the method

Since the proposed implementation solves the Poisson equation over the whole image domain instead of the selected sub-domain, it is a sanity check to verify that the known pixel values are only
minimally altered. This was done experimentally with the original images of Figs. 4, 6 and 7. On these images we solved the Neumann problem

$$\begin{cases} \Delta u = \Delta f, & \text{in } \mathbb{R}, \\ \frac{\partial u}{\partial n} = 0, & \text{in } \partial \mathbb{R} \end{cases} \tag{18}$$

and finally we imposed that mean and standard deviation of $u$ remain the same as for $f$. Table 1 shows the MSE, PSNR, and MD (maximal difference) between $u$ and $f$. It remains far below the any perceptual difference in these images.

To confirm that outside the selection area the changes introduced are negligible and no Gibbs effect is produced, Fig. 9 shows an experiment where we have pasted a high frequency texture on a low frequency texture. The Poisson editing technique creates a minimal perturbation on the background and no Gibbs effect. By looking carefully at the boundary of the pasted region, one can, however observe some diffusion from the extrema of the high frequency texture into the low texture region.

### 6. Discussion

Seamless image editing introduced by Pérez et al. (2003) requires to solve a large linear system on an arbitrary region of the image domain, and becomes slow and difficult if the region is non connected and complex, thus not amenable to multigrid methods. The Fourier approach proposed here is able to solve all image editing tools proposed in (Pérez et al., 2003). The Fourier method is exact, non-iterative, and allows to treat regions with arbitrary topology. Its main advantage with respect to former implementations is that it is no more limited to a manual selection of one or several simple regions; image regions selected automatically by any criterion can be edited at no additional computational cost. There are only two cases where the original numerical method by finite difference and the multigrid decomposition is preferable. The first case is when the edited region is actually small and rectangular, in which case the complexity of a multigrid method, being proportional to the domain size, becomes inferior to the FFT complexity on the whole image. The second case is when it is absolutely required not to alter, even minimally, the colors in the image outside the edited zone. To illustrate the simplicity of the method, a fast and automatic local contrast enhancement algorithm treating complex regions has been tested.
References


