A Spectral Retouching Tool for Removing Repetitive Image Noise

Hannes Jung, Wilhelm Burger

Upper Austria University of Applied Sciences, School of Informatics, Communications and Media
Digital Media Department, Softwarepark 11, A-4232 Hagenberg, Austria
Email: wilbur@ieee.org

Abstract – This paper describes a method and implementation for removing global, repetitive 2D artifacts in images by interactively editing their Fourier spectra. Based on this concept, “InSpectral” is an interactive software application that provides intuitive and easy-to-use image editing functionality in the spectral domain with immediate visual feedback in the spatial domain.

Keywords – Discrete Fourier transform, repetitive pattern removal, denoising, frequency domain editing, image retouching, Java, ImageJ.

1 INTRODUCTION

Pictures have gained major relevance in our daily lives. They do not just capture important moments but also transport feelings and emotions. In former times, people used their analog cameras only for special occasions but, especially since the introduction of digital cameras, images made their way into everyday life and have become a social commodity. Of course, images of these special moments should be of high quality. Therefore users try to improve them by editing with applications like Adobe Photoshop Lightroom1 or Apple Aperture.2 These applications provide various tools (e.g., noise removal, red eye correction, sharpening, etc.) to help users improve the quality of their pictures.

Unfortunately, there is no easy solution for every problem available. For example, it is very difficult to remove the printing raster of a newspaper scan, as shown in Fig. 1. This paper deals with an intuitive approach to correct such repetitive patterns from images, based on their distinct effects in the frequency domain.

To cope with the appearance of such periodic interferences, it is useful to know that repetitive image components correspond to local energy peaks in the image’s Fourier spectrum. These energy peaks show up as bright spots in the corresponding power spectrum, as shown in Fig. 1 (b).

As the next step, the energy peaks caused by the periodic noise pattern are erased in order to annihilate them in the spatial domain. Once the modifications in the Fourier spectrum are done, the inverse 2D DFT,

\[ g(u,v) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m,n) \cdot e^{2\pi i \left( \frac{um}{M} + \frac{vn}{N} \right)} , \]

is applied to obtain the filtered version of the input image.

1.1 Image padding

With the FFT it is possible to reduce the time complexity of the computation with an input signal of length \( N \) from \( \Theta(N^2) \) to only \( \Theta(N \cdot \log_2 N) \). FFT algorithms are optimized for input signal lengths of \( 2^k \), for \( k \in \mathbb{N} \). Therefore, pictures whose width or height are not powers of two must be either cropped or enlarged appropriately. Figure 2 shows two different ways to accomplish this. Method (a) stretches the image content to a size of a power of two, but because of the implicit periodicity of the input signal/image, there may appear large intensity differences (signal energy) at the image borders. A common solution to

---

1http://www.adobe.com/products/photoshoplightroom/
2http://www.apple.com/aperture/
This problem is to “pad” the image by applying a suitable windowing function, such as the Hanning function, \( w(x) = \cos^2 \left( \frac{x}{D} \pi \right) \), where \( D \) is the width of the window (Fig. 3). Thereby the transition from image content to image background is smoothed, as shown in Fig. 2 (b).

### 1.2 Symmetry and periodicity

One of the most important properties of the DFT is the symmetry of the resulting spectrum for real-valued input data (images are usually real-valued). Hence it is possible to mirror refinement steps with respect to the spectrum’s origin, as shown in Fig. 4, and thereby to reduce the required time and effort needed to edit the power spectrum.

To compute the Fourier transform or the resulting Fourier spectrum numerically on a computer, a discrete and periodic input signal is needed. Only such an input generates a discrete and finite spectrum. In other words, even though the Fourier spectrum is computed from a finite-length sequence (vector), the underlying signal is implicitly treated as being periodic. In the 2D case, the discrete spectrum represents an infinite image that is periodic in both dimensions. As a consequence of this truncation, the possible intensity steps at the transitions between left/right and top/bottom image boundaries generally induce strong signal energy over a wide frequency range. A common solution to this problem is the use of windowing functions, as described in [2, 3].

### 1.3 Linear filtering in the frequency domain

One of the main application areas of the Fourier transform, especially the fast Fourier Transform, in the field of image processing is the implementation of linear filters. The main advantage of using the Fourier transform for filtering images is the computational efficiency of applying filters of larger sizes.

This is due to the well-known convolution property of the Fourier transform, meaning that a linear convolution in the spatial domain between a signal \( g \) and a filter kernel \( h \), denoted \( g * h \), corresponds to a point-wise multiplication of the corresponding spectra in the frequency domain, \( G \cdot H \). Here \( G, H \) are the Fourier transforms of \( g, h \), respectively. This process can be illustrated in the following way:

\[
\begin{align*}
g(u,v) \ast h(u,v) &= g'(u,v) \\
\text{DFT} & \quad \text{DFT} \\
G(m,n) \cdot H(m,n) & \rightarrow G'(m,n)
\end{align*}
\]

First, the input image \( g(u,v) \) and the filter function \( h(u,v) \) get transformed to the frequency domain by applying the forward DFT. The linear filter operation is implemented as a point-wise (complex) multiplication in the frequency domain. The modified spectrum \( G'(m,n) \) is transformed back to the spatial domain by applying the inverse DFT. The result of this transformation is the filtered version of the input image \( g'(u,v) \). Considering the fact that the convolution of an image of size \( M \times M \) with a filter kernel of size \( N \times N \) has a time complexity of \( \mathcal{O}(M^2N^2) \), but
only $\Theta(M \log_2 M)$ when using a fast Fourier Transform, we can say that it is almost a necessity to use a FFT for filtering with large kernels because of the enormous time saving.

This application uses the multiplication of a complex-valued Fourier spectrum $G(m,n) \in \mathbb{C}$ with a real-valued transfer function $H(m,n) \in \mathbb{R}$,

$$
\tilde{G}(m,n) = \begin{bmatrix}
G_{Re}(m,n) \\
G_{Im}(m,n)
\end{bmatrix} = \begin{bmatrix}
G_{Re}(m,n) \\
G_{Im}(m,n)
\end{bmatrix} \cdot H(m,n)
$$

to apply a filter to a spectrum. Hereby the real and imaginary part of the spectrum are multiplied with the same filter coefficient and thus the phase angle $\phi$ of the resulting vector is not changed.

2 IMPLEMENTATION

The main purpose of this work was the implementation of an image processing application which provides an intuitive approach to retouching images in the frequency domain. The application has been implemented in Java using the Eclipse\(^3\) environment. All the management of the image data is controlled by the use of ImageJ which is a free Java-based image processing library from the National Institutes of Health\(^5\). Furthermore, two available implementations\(^4\) were used for computing the forward and inverse FFT. Thereby, the classes of both algorithms were adapted accordingly. InSpectral currently only operates on grayscale images – color images are automatically converted to grayscale after opening.

2.1 User interface

Figure 5 shows the graphical user interface of the InSpectral tool. It consists of three main parts:

- working area,
- toolbars (main menu and toolset),
- adjustments panel.

All opened images are shown in the working area. From each of them the power spectrum is computed and displayed instantly. Furthermore, it is possible to edit any power spectrum in this window by simply painting in it with one of several drawing tools. This works just like every other image editing software, such as Adobe Photoshop\(^5\) or CorelDraw.\(^6\) Both tool bars provide functional editing instruments that ease the way of image retouching to a large extent. For example, the “symmetrical editing” option allows the user to suppress one energy peak and to mirror the editing step. In addition, it is possible to adjust almost every available tool by changing properties in the adjustment panel (e.g., filter radius and weights).

2.2 Interaction

The interactive editing cycle can be summarized as follows:

1. After opening the image, the Fourier spectrum of the input image is calculated using the fast Fourier transform. The spectrum is generally complex-valued, which means that each coefficient consists of a real and an imaginary part.

2. Since it is difficult to visualize the complex-valued Fourier spectrum, both parts (real and imaginary) are combined into the power spectrum, which is real-valued and positive and can thus be displayed easily. Note that the power spectrum itself is not used for editing; it is only a visual aid to illustrate the structure of the Fourier spectrum. The actual editing process is done in the original (complex-valued) Fourier spectrum.

3. A weight image is generated in the background when a user applies one of the provided tools to the power spectrum.

4. After each editing step (triggered by releasing the mouse button), this weight image is combined with the real and imaginary parts of the Fourier spectrum by a point-wise multiplication.

5. Finally, the filtered input image is reconstructed from the updated power spectrum by an inverse FFT. Thereby, the user receives immediate feedback on how the spectral filter operation affects the image in the spatial domain.

2.3 Implemented filters

To cope with all characteristics of a Fourier spectrum, several types of linear filters (transfer functions) were implemented in this tool: quadratic, circular, conic, cosine, Hanning, Hamming, and Gaussian. All these filters can be adapted to the individual properties of the given spectrum by adjusting the corresponding parameters, like the radius (bandwidth of the filter) or

\(^3\)http://www.eclipse.org
\(^5\)http://www.adobe.com/photoshop
\(^6\)http://www.coreldraw.com
Fig. 6. Picture of a car with very strong and rough print raster (see original in Fig. 1. The resulting image after applying a Gaussian filter (a) is much finer and clearer. The edited power spectrum is shown in (b). The image is only a small cutout of the original.

the strength of application (attenuation). Furthermore there are two different tools available to apply any of these spectral filters. First, there is the “brush” tool which enables the user to apply a filter at an exact location or arranged along a custom brush stroke. The alternative is the “line” tool, which applies the current filter along a straight line in the spectrum with a specified width.

3 RESULTS

The approach of the image enhancement and the quality of the results are best explained by some test cases.

3.1 Print raster removal

Figure 6 (a) shows a cutout of an image that shows a car with a very rough printer raster (see the original in Fig. 1 (a)). It is very unattractive to use such a picture for digital work. The periodic printer raster manifests itself as multiple energy peaks symmetrically located around the origin of the power spectrum. To minimize the impacts caused by these peaks, a filter with a Gaussian transfer function was applied with double symmetry. Although the raster is still visible after the editing, the image is now much clearer and small details seem to have been enhanced. In addition to the refinement of the raster, the image got smoothed too.

3.2 Interlaced video

Taking single stills from a typical video sequence is a notorious problem due to the interlaced scanning scheme. The distracting scan lines displayed in figure 7 (a) show up in the power spectrum as bright fields. After erasing them with a Gaussian filter and the line tool they completely disappear in the resulting image. Furthermore, it can be observed that static, not moving image parts (e.g., books in the background) are not affected by the spectral retouching.

Fig. 7. Video still of a moving foot with strong “interlacing” artifacts (a) and its power spectrum (b). The filtered image (c) and the edited power spectrum (d).

4 SUMMARY

The use of Fourier analysis for removing global, repetitive artifacts is an obvious decision. The difficulty is to design suitable filters in the frequency domain. We have proposed a technique where frequency-domain filters are applied interactively and incrementally with immediate visual feedback. After the analysis of several test images and the achieved results it is obvious that image retouching in the frequency domain is very efficient at removing repetitive image noise. Moreover, the execution time is short and feedback appears almost in real time, even when working with larger images. The “InSpectral” tool is well suited for spectral image retouching because there is no preliminary knowledge required to use the application and good results can be achieved in a very short amount of time.

REFERENCES


