Real-Time Frequency Adjustment of Images and Videos

Rafael L. Germano*
Institute of Informatics
Federal University of Rio Grande do Sul
Porto Alegre, RS, Brazil
Email: rlgermano@inf.ufrgs.br

Eduardo S. L. Gastal
Institute of Informatics
Federal University of Rio Grande do Sul
Porto Alegre, RS, Brazil
Email: eslgastal@inf.ufrgs.br

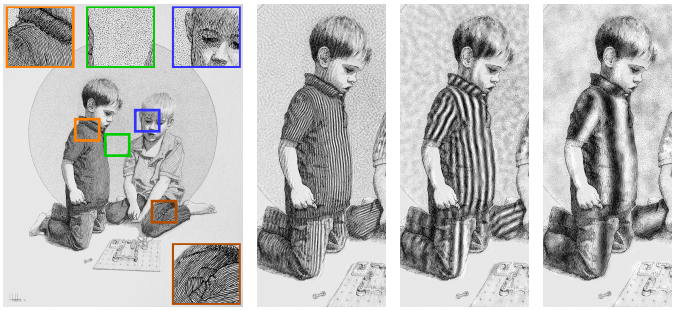
***Abstract*—We present a technique for real-time adjustment of spatial frequencies in images and videos. Our method allows for both decreasing and increasing of frequencies, and is orthogonal to image resizing. Thus, it can be used to automatically adjust spatial frequencies to preserve the appearance of structured patterns during image downscaling and upscaling. By pre-computing the image's space-frequency decomposition and its unwrapped phases, these operations can be performed in real time, thanks to our novel mathematical perspective on frequency manipulation of digital images: interpreting the problem through the theory of instantaneous frequencies and phase unwrapping. To make this possible, we introduce an algorithm for the simultaneous phase unwrapping of several unordered frequency components, which also deals with the frequency-sign ambiguity of real signals. As such, our method provides theoretical and practical improvements to the concept of spectral remapping, enabling real-time performance and improved color handling. We demonstrate its effectiveness on video and a large number of images subject to frequency adjustment. By providing real-time control over the spatial frequencies associated with structured patterns, our technique expands the range of creative and technical possibilities for image and video processing.**

I. Introduction

We present a new mathematical perspective on frequency manipulation of digital images and videos. Using the theory of instantaneous frequencies and phase unwrapping operators [1], our **real-time spatial frequency adjustment technique** is the first method capable of performing image edits such as the ones shown in Figure 1 in real time. The performance improvement resulting from our formulation is also critical for making frequency adjustments in videos practical.[1] The idea of our method is to decompose an image into local phase and amplitude components such that the phase encodes the high-frequency structured content. This decomposition provides simple and interesting ways of manipulating frequency through phase manipulation.

Our technique starts by obtaining a space-frequency decomposition of the image (orange block of Figure 2). The decomposition represents the original image as a sum of several cosine functions and a residual. Each cosine has an associated frequency and amplitude. The frequencies are jointly integrated by our weighted least-squares system (blue block of Figure 2) to obtain smooth phase functions that may be scaled by a factor α to reconstruct the image with the original frequencies adjusted by α (green block of Figure 2). The reconstruction algorithm may be performed per-pixel, allowing for a fast parallel real-time implementation. Despite working with cosine functions for the decomposition, our technique is able to handle complex structured patterns, as demonstrated in Figure 5.

Our novel mathematical framework is fundamental in allowing our approach to work in real time. To make this possible, we introduce technical contributions to ridge analysis and 2-D / 3-D phase unwrapping of digital images and videos (Section IV-B).



(a) Reference image (b) Our $\alpha = 0.3$ (c) Our $\alpha = 0.1$ (d) Our $\alpha = 0.03$

(e) Our (**0.01 s**) (f) SR (**1.5 s**) (g) O-MOMS (0.01 s)

Fig. 1. (a) Reference image with 1,618 × 2,048 pixels (hand drawing "Brothers", by artist Tyler Hobbs, used with permission). (b–d) Examples of frequency adjustments applied to (a) using our technique, with the images kept at the original resolution (please zoom-in to see the details and to avoid aliasing artifacts caused by the PDF reader's resampling filter). (e) Downscaled version of (a) generated with our technique in real time (0.01 seconds for frequency adjustment and image reconstruction). (f) Downscaled version of (a) obtained with spectral remapping (SR) [2] (1.5 seconds). (g) Downscaled version of (a) without changing the image frequencies. Note how the high-frequency patterns from the original image are lost. In images (e–g), resampling to 135 × 171 pixels was done using the O-MOMS method [3].

* M.Sc. dissertation: https://lume.ufrgs.br/handle/10183/225714.

[1] Please refer to the following link for some videos demonstrating our work: https://www.inf.ufrgs.br/~eslgastal/RealTimeFrequencyAdjustment/

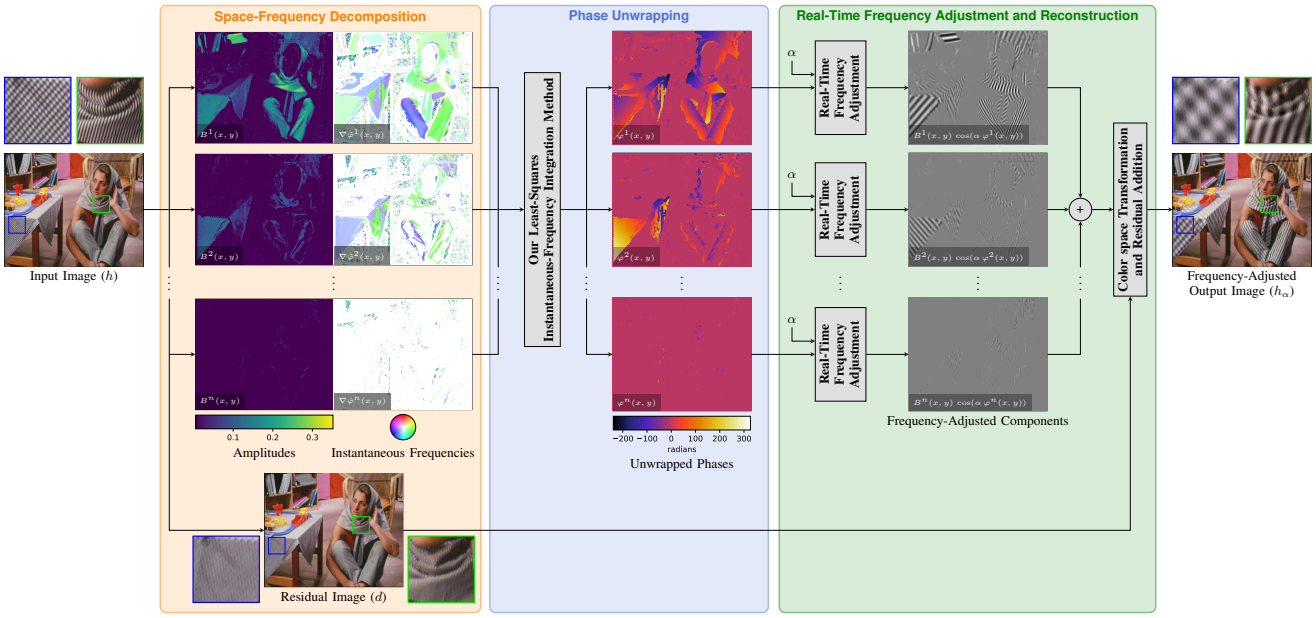


Fig. 2. Overview of our frequency-adjustment technique. *(Orange block)* The input image is decomposed into several cosine waves (Eq. (4)), represented by per-pixel scalar amplitudes and instantaneous-frequency vectors. The frequency vector direction and magnitude are given by the vector from the center of the color wheel to the pixel color in the wheel. This decomposition may be performed in the RGB space (applied to each color channel independently) or in some other color space where channels better encode the high-frequency content, such as the PCA-defined color space used for this illustration. *(Blue block)* The detected frequencies are then integrated by our method to recover a continuous scalar field of unwrapped phases. Note that the unwrapped phase values for each pixel are not sorted between the images, and as such the phase data may appear more discontinuous than they actually are. Note also the absence of phase information for low-detail regions, which are treated as residue only. *(Green block)* The frequency-adjusted content is obtained by reconstructing the waves from their original amplitudes and corresponding scaled unwrapped phases (Eq. (19)), using a user-provided scaling factor α (for this example, $\alpha = 0.25$). All waves are summed to obtain the frequency-adjusted content. Finally, to obtain the reconstructed image, the frequency-adjusted content is transformed back to RGB space (if initially in RGB space no transformation is needed) and summed back to the residual.

We also show how to handle the problematic sign-ambiguity that arises from the conjugate-symmetry of real waves (Section IV-A1). Our approach also improves color handling when compared to recent frequency remapping methods [2], by correlating the color channels during frequency integration (Section IV-C). This improvement is visible in Figure 6.

By defining the frequency scaling factor α as a function of an image resizing factor r, spatial frequencies are automatically adjusted to preserve the appearance of structured patterns during image downscaling (Figure 1(e)). Combined with our method's real-time performance, this opens up a variety of interesting applications, such as dynamically computing detail-preserving thumbnails for image galleries, or adapting an image's content in real time based on viewing distance.

In summary, the **contributions** of our work include:

- *The first real-time technique for spatial frequency adjustment of images and videos* (Section III). It is based on a novel mathematical framework that enables its improved performance over the state of the art. It requires preprocessing the space-frequency decomposition and the unwrapped phase of the image, but frequency adjustment is subsequently performed in real-time.
- *A phase-unwrapping technique to recover the phases of real waves* that locally reconstruct an image (Section IV). Our solution naturally handles a sign ambiguity involving the phase unwrapping of real waves, in addition to simultaneously treating many unordered phase components without explicit sorting. As such, it extends the state-of-the-art on phase unwrapping that only covers a single phase component wrapped to $[-\pi, \pi)$.
- *A technique for processing color images for use with spatial frequency adjustment* (Section IV-C). It is based on a multi-channel optimization performed in RGB space and avoids some color artifacts associated with the use of PCA-based color reconstruction.

II. Related Works and Background

Our work was inspired by the *spectral remapping* (SR) technique for image downscaling [2]. SR remaps frequencies that are not representable in the target resolution to new representable frequencies, preserving the orientations of the original high-frequency patterns. This solution effectively encodes important structured details at lower frequencies during downscaling, instead of discarding them with an anti-aliasing filter. This results in more faithful representations of the original content. The technique, however, has only been demonstrated for image downscaling in combination with a remapping function that takes similarly-oriented waves to a common frequency. More importantly, *(i)* it is *non-interactive*, requiring the solution of a sparse linear system for each combination of input image I and downscaling factor r; and *(ii)* the PCA-based reconstruction strategy used for processing color images may introduce some noticeable artifacts depending on the target resolution.

Our frequency-adjustment technique can be employed to image downscaling and addresses the aforementioned issues being up to $100\times$ faster than SR. In addition, we also demonstrate its extension to videos.

To achieve our results, we introduce a *phase unwrapping* algorithm that simultaneously handles several real-wave phase components, in addition to inter-component dependencies (Section IV). Phase unwrapping is the process of recovering a continuous function $\theta(x)$ from samples $\hat{\theta}(x)$ whose values are only known up to modulo 2π [4]. This procedure is a fundamental building block in many applications, ranging from satellite topography and marine/earth seismology to the analysis of brain waves [5], [6]. Phase unwrapping is tightly linked to the theory of instantaneous-frequency decompositions and space/time-frequency analysis [1]. In this context, it is often called "ridge" analysis, and it may be derived from Gabor [7] and wavelet transforms [8]. There is a significant amount of theory and algorithms developed in this area for 1-D signals, such as acoustic and speech data [9]. The corresponding 2-D and 3-D problems (required for our frequency-adjustment technique) are significantly more challenging to solve [10].

III. Frequency Adjustment

Consider a continuous wave signal $s(x,y) = \cos(\theta(x,y))$ described by a continuously-differentiable phase function θ. The horizontal $\nabla_x\theta(x,y)$ and vertical $\nabla_y\theta(x,y)$ *instantaneous frequencies* of this signal are given by the partial derivatives of its phase [1]:

$$\nabla_x\theta(x,y) = \frac{\partial}{\partial x}\theta(x,y) \quad \text{and} \quad \nabla_y\theta(x,y) = \frac{\partial}{\partial y}\theta(x,y). \tag{1}$$

Similarly, the local wavefront orientation (normal direction) at (x,y) is given by the gradient vector

$$\nabla\theta(x,y) = \left[\nabla_x\theta(x,y), \nabla_y\theta(x,y)\right].$$

A *frequency-adjusted* version s_α of s, where frequencies are scaled by a factor $\alpha > 0$, is obtained as:

$$s_\alpha(x,y) = \cos(\alpha\theta(x,y)). \tag{2}$$

It is easy to see that the horizontal and vertical frequencies of s_α are $\alpha\nabla_x\theta$ and $\alpha\nabla_y\theta$, i.e., scaled versions of the corresponding frequencies of s. The local direction of wave propagation is preserved in s_α since $\nabla\alpha\theta = \alpha\nabla\theta$. Finally, s_α is still a continuous function, as was s. Figure 3 illustrates an example of frequency adjustment for a signal where θ is known analytically.

A. Frequency Adjustment of Natural Images

Given a natural image $h(x,y)$, we seek a decomposition which represents it as a sum of N components with phase functions $\theta^1 \dots \theta^N$, amplitude functions $A^1 \dots A^N$, and a residual $d(x,y)$:

$$h = s + d \quad \text{where} \quad s(x,y) = \sum_{n=1}^{N} A^n(x,y)\cos(\theta^n(x,y)). \tag{3}$$

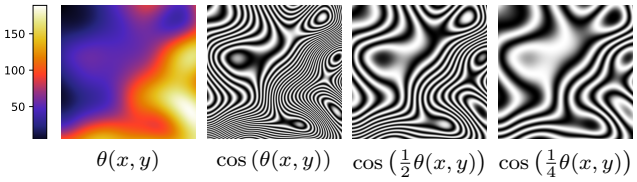


Fig. 3. Example of frequency adjustment. The continuous phase function $\theta(x,y)$ is defined by a Perlin noise scalar field (left). Scaling the phase by factors $\alpha > 0$ results in frequency-adjusted signals $\cos(\alpha\theta(x,y))$, which preserve the local wavefront orientations.

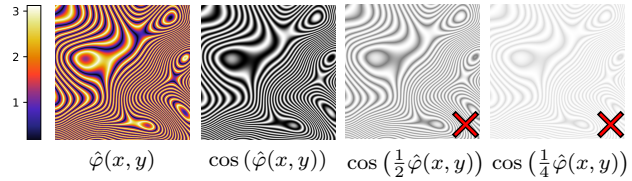


Fig. 4. Example of unsuccessful frequency adjustment. When the phase function $\hat{\varphi}(x,y)$ is wrapped (values recovered as $\hat{\varphi} = \arccos(\cos(\varphi))$ are in $[0,\pi]$ – see color scale on the left), scaling the phase by factors $\alpha > 0$ does **not** result in frequency-adjusted signals. One must therefore **unwrap** the phase function before being able to perform frequency adjustment. Compare to Figure 3.

Obtaining such a decomposition for a given image h is a severely under-constrained problem. Thus, we require the summation in $s(x,y)$ to only encode high-frequency content that should be affected by the frequency adjustment (remaining content goes into the residual d). Furthermore, the phase functions are required to be piecewise smooth for the instantaneous frequencies to be well defined almost everywhere, making $s(x,y)$ (and thus h) amenable to frequency adjustment. Finally, the amplitudes $A^n(x,y)$ should vary slowly when compared to local frequencies, so that high-frequency variations become encoded in the phases only [1].

B. Finding a Decomposition

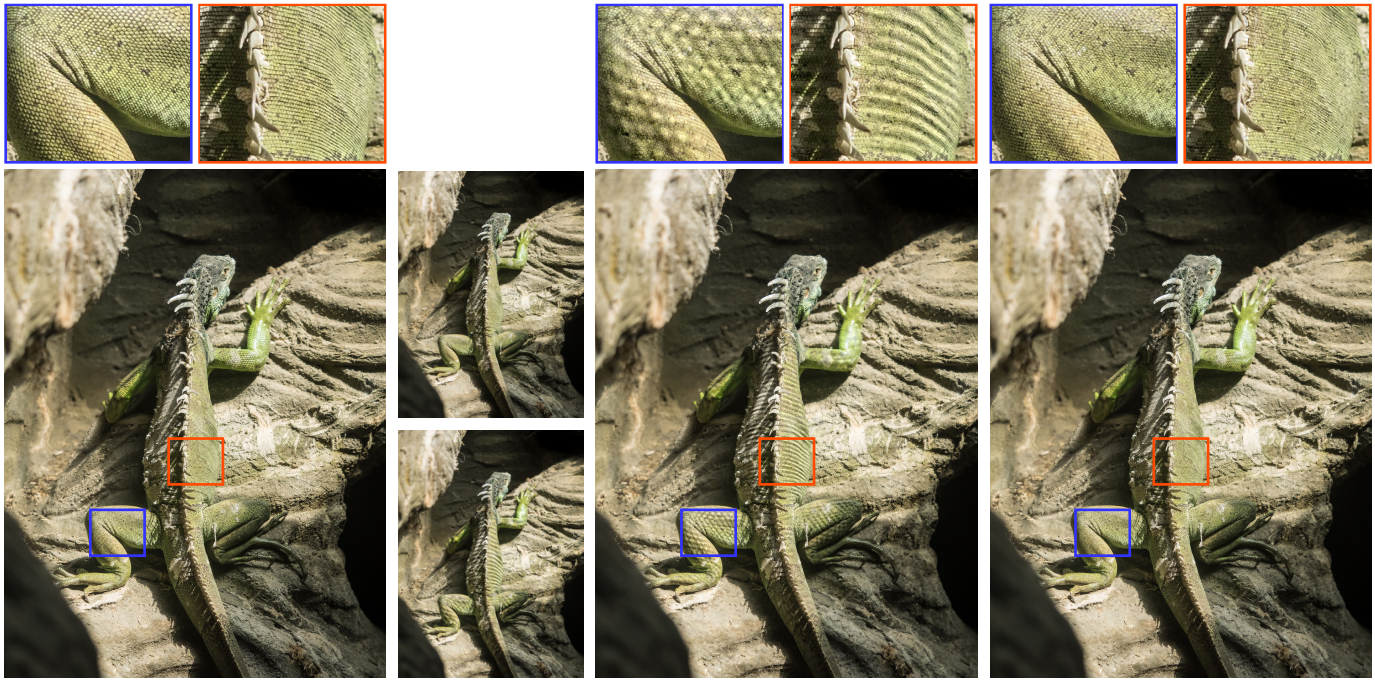
Time/Space-frequency analysis may be employed to measure the instantanous frequencies of several non-harmonic spectral components [1]. Gastal and Oliveira [2] describe a practical algorithm for this purpose, approximating the neighborhood around each pixel as a sum of Gaussian-windowed plane waves. From the constant frequencies and phase-shifts of such waves it is possible to extract, for each pixel, a series of phase values $\hat{\varphi}^n(x,y)$ and associated amplitudes $B^n(x,y)$, $n = 1 \dots N$, such that

$$h = s + d \quad \text{where} \quad s(x,y) = \sum_{n=1}^{N} B^n(x,y)\cos(\hat{\varphi}^n(x,y)). \tag{4}$$

Despite the similarity of this expression to that of Eq. (3), the values $\hat{\varphi}^n$ cannot be directly employed for frequency adjustment since they are *wrapped* to $[0,\pi]$. The problem that occurs when trying to do so is illustrated in Figure 4.

IV. Phase Unwrapping in Two Dimensions

While 1-D phase unwrapping has a simple solution [12], 2-D unwrapping is more involving and requires specialized



(a) Reference (1,536 × 2,048) (b) ($r = \alpha = 0.14$) (c) Ours ($r = 1$, $\alpha = 0.25$) (d) Ours ($r = 1$, $\alpha = 1.25$)

Fig. 5. Picture of a lizard (courtesy of William Warby). (a) Reference image. (b) Downscaled version of (a) generated with (bottom) and without (top) our technique for frequency adjustment and resampling to 220 × 293 using the method of [11] and [3], respectively. (c) and (d) Examples of spatial frequency adjustment generated by our technique. The images were kept at the original resolution ($r = 1$): (c) Frequency decreasing obtained with a scaling factor $\alpha = 0.25$. (d) Frequency increasing obtained with a scaling factor $\alpha = 1.25$. Note the adjusted frequencies of the details on legs and belly of the lizard. (Please zoom-in to see the details and to avoid aliasing artifacts caused by the PDF reader's resampling filter).

algorithms [10]. *These algorithms work with a wrapped phase* $\hat{\theta} \in [-\pi, \pi)$. Thus, solving the phase unwrapping problem in this case consists only in finding the shift factors $k(x, y)$ in Eq. (5) (that is, there is no sign-ambiguity problem).

Digital images are real functions and modeled using real waves: the cosines in Eq. (3). Therefore, in this situation the measured phase *has a reduced angular range*, $\hat{\varphi} = \arccos(\cos(\theta)) \in [0, \pi]$, and an *ambiguous sign*, since $\cos(\hat{\varphi}) = \cos(-\hat{\varphi})$. As a result, $\hat{\varphi}$ is related to the true phase θ by an unknown sign, σ, in addition to an unknown shift, k:

$$\theta(x, y) = \sigma(x, y)\, \hat{\varphi}(x, y) + 2\pi k(x, y) \quad \text{for} \quad \sigma(x, y) \in \{-1, +1\}. \tag{5}$$

Such *sign ambiguity* and *reduced angular range* make phase unwrapping for digital images and videos a challenging task.

A. Unwrapping the phase of one real wave

For frequency adjustment of digital images, one is interested in unwrapping the phase $\theta(x, y)$ from a real wave component $\cos(\theta(x, y))$. The wrapped phase measurement $\hat{\varphi}(x, y) \in [0, \pi]$ is thus related to the true phase according to Eq. (5). Since $k(x, y)$ defines a piecewise-constant function, the unwrapped and wrapped gradients are related only by the unknown sign $\sigma(x, y)$ at each position (x, y):

$$\nabla\theta(x, y) = \sigma(x, y)\nabla\hat{\varphi}(x, y) \quad \text{almost everywhere.} \tag{6}$$

We can obtain $\nabla\hat{\varphi}$ directly from a space-frequency decomposition. We now focus on how to integrate the gradient field $\nabla\theta = \sigma\nabla\hat{\varphi}$ to recover the phase function, *without explictly computing the signs* σ.

(Throughout the text we denote the pointwise product of two functions f and g as fg; that is, $(fg)(x, y) = f(x, y)\, g(x, y)$. Furthermore, $f = g$ denotes pointwise equality: $f(x, y) = g(x, y), \forall (x, y)$.)

1) Integrating the phase gradient field for one real wave: To recover the unwrapped phase θ, the gradient field $\nabla\theta = \sigma\nabla\hat{\varphi}$ is integrated in the least-squares sense.

Let T_x and T_y be two unit-shift operators that act on a function θ as $T_x(\theta)(x, y) = \theta(x-1, y)$, and $T_y(\theta)(x, y) = \theta(x, y-1)$. Furthermore, let $\circ$ in $T_\circ$ stand for either x or y. Then, a trapezoidal integration rule gives the following linear relations between the unknown phase values $\theta(x, y)$ of all pixels [13]:

$$\theta - T_\circ(\theta) = \tfrac{1}{2}\big[\, \nabla_\circ\theta + T_\circ(\nabla_\circ\theta) \,\big]. \tag{7}$$

Given that $\nabla\theta = \sigma\nabla\hat{\varphi}$, Eq. (7) may be expanded as (we underline the parts of the equations that have changed):

$$\theta - T_\circ(\theta) = \tfrac{1}{2}\big[\, \underline{\sigma\nabla_\circ\hat{\varphi}} + T_\circ(\underline{\sigma\nabla_\circ\hat{\varphi}}) \,\big]. \tag{8}$$

By introducing a function φ satisfying $\sigma\varphi \overset{\text{def}}{=} \theta$, Eq. (8) becomes

$$\underline{\sigma\varphi} - T_\circ(\underline{\sigma\varphi}) = \tfrac{1}{2}\big[\, \sigma\, \nabla_\circ\hat{\varphi} + T_\circ(\sigma\, \nabla_\circ\hat{\varphi}) \,\big]. \tag{9}$$

Note that to solve the above equation one must know σ. To remove this dependency, we start by multiplying both sides by σ:

$$\underline{\sigma^2}\varphi - \underline{\sigma} T_\circ(\sigma\varphi) = \tfrac{1}{2}\big[\, \underline{\sigma^2}\, \nabla_\circ\hat{\varphi} + \underline{\sigma} T_\circ(\sigma\, \nabla_\circ\hat{\varphi}) \,\big]. \tag{10}$$

Note that $\sigma^2 = 1$ for any (x, y), since $\sigma(x, y) \in \{-1, +1\}$. Furthermore, since $T_\circ(fg) = T_\circ(f)T_\circ(g)$, Eq. (10) simplifies to

$$\varphi - \underline{\Delta_\circ \sigma}\, T_\circ(\varphi) = \tfrac{1}{2}\big[\, \nabla_\circ \hat{\varphi} + \underline{\Delta_\circ \sigma}\, T_\circ(\nabla_\circ \hat{\varphi})\,\big], \qquad (11)$$

where $\Delta_\circ \sigma \stackrel{\text{def}}{=} \sigma T_\circ(\sigma)$. Note that *$\Delta_x\sigma$ and $\Delta_y\sigma$ encode the locations where σ changes sign*. More precisely, $\Delta_x\sigma(x, y) = -1$ whenever the sign of $\sigma(x, y)$ differs from the sign of the neighboring pixel $\sigma(x-1, y)$, and $\Delta_x\sigma(x, y) = +1$ otherwise. An analogous relation is true for $\Delta_y\sigma(x, y)$ in the y-dimension.

As shown below in Section IV-A2, it is possible to compute $\Delta_\circ\sigma$ despite σ being unknown. As such, the only unknowns in Eq. (11) are the values of the function φ, which may be found by solving the associated linear system through least squares:

$$\min_{\varphi} \sum_{\circ \in \{x,y\}} \big\| \varphi - \Delta_\circ \sigma T_\circ(\varphi) - f_\circ \big\|^2 + \lambda \|\varphi\|^2, \qquad (12)$$

where $f_\circ(x, y)$ encodes the right-hand-side of Eq. (11), for $\circ \in \{x, y\}$. The recovered phase φ is enough to perform frequency adjustment, without knowing the per-pixel sign σ, since the cosine is an even function. Thus $\cos(\alpha\,\varphi) = \cos(\alpha\,\sigma\,\varphi) = \cos(\alpha\,\theta)$ and so it is not necessary to recover θ.

Eq. (12) has a unique solution obtainable from the linear system

$$(D_x^T D_x + D_y^T D_y + \lambda\, I)\,\varphi = D_x^T f_x + D_y^T f_y, \qquad (13)$$

where I is the identity matrix, and D_x and D_y are similar to backward-difference matrices, but *(i)* have some entries with flipped signs (according to $\Delta_x\sigma$ and $\Delta_y\sigma$); and *(ii)* only include contraints for non-missing values (the reconstruction of pixels with missing values is handled by windowing, discussed in Section V). We use $\lambda = 10^{-6}$ in the regularizer since it produces good results according to our empirical tests.

2) Determining sign changes and computing $\Delta\sigma$: Let p and q be two neighboring pixels, with local gradients $\nabla\hat{\varphi}(p)$ and $\nabla\hat{\varphi}(q)$. Also, recall that $\nabla\theta = \sigma\nabla\hat{\varphi}$. Two situations are possible:

1) **There is** a sign change between the phase gradients $\nabla\theta(p)$ and $\nabla\theta(q)$ of p and q, meaning that $\sigma(p) \neq \sigma(q)$;
2) **There is no** sign change between p and q, and $\sigma(p) = \sigma(q)$.

Possibility (1) implies $\Delta\sigma(p) = \sigma(p)\,\sigma(q) = -1$, while possibility (2) implies $\Delta\sigma(p) = \sigma(p)\,\sigma(q) = +1$. To determine which one is true, we recall that the phase θ should be continuously differentiable, meaning that the phase gradient $\nabla\theta$ should be continuous. As such, we choose the option that better preserves the continuity of $\nabla\theta$, i.e., that results in the smallest change between $\nabla\theta(p)$ and $\nabla\theta(q)$. Thus,

$$\Delta\sigma(p) = \underset{v \in \{-1,+1\}}{\arg\min} \big\| \nabla\hat{\varphi}(p) - v\,\nabla\hat{\varphi}(q) \big\|^2. \qquad (14)$$

B. Unwrapping the phases of several real waves

Natural images contain complex structured patterns. These patterns require the summation of several cosine components in order to be correctly represented by the decomposition of Eq. (3). The several phase functions $\{\theta^1, \theta^2, \ldots\}$ that build-up a pattern are in principle independent and could be separately unwrapped. In practice, however, this independent unwrapping is severely error prone. To understand why, note that, before unwrapping, one would be required to assign each n-th gradient measurement $\nabla\hat{\varphi}^n(p)$, of a pixel p, to a particular phase component $\theta^{\kappa_p(n)}$, where $\kappa_p(n)$ is an indexing function (permutation) specific for p. Furthermore, the permutations κ_p and κ_q for all neighboring pixels p and q should be consistent to each other, assigning the gradients that belong together to the same phase component. Otherwise, severe unwrapping errors will occur.

Due to ambiguous information returned by the space-frequency decomposition, sometimes it is not possible to precisely define the permutations κ_p for all pixels p. Furthermore, for particular pixels, the permutations are often incorrectly defined due to noise or frequency measurement errors. As a result, the whole phase unwrapping solution becomes compromised.

We avoid the aforementioned difficulties by, instead, defining weighted associations between *all* measurements $\nabla\hat{\varphi}^n$ and *all* phase functions θ^n (in place of discrete permutations). This significantly reduces the possibility of unwrapping errors, especially in the presence of ambiguous information returned by the space-frequency decomposition. It also makes the algorithm more robust to noise and measurement errors, and handles missing phase information.

Consider a pixel p and let $N(p)$ denote the number of measured phase components for p. Furthermore, let $T_\circ(p)$ denote the neighboring pixel of p according to the unit-shift operator $T_\circ$. For valid indices $n = 1 \ldots N(p)$ and $m = 1 \ldots N(T_\circ(p))$, the inter-component version of the integration rule (Eq. (11)), which couples the n-th component of pixel p to the m-th component of the neighboring pixel $T_\circ(p)$, is given by

$$\varphi^n - \Delta_\circ^{nm}\sigma\, T_\circ(\varphi^m) = \tfrac{1}{2}\big[\, \nabla_\circ\hat{\varphi}^n + \Delta_\circ^{nm}\sigma\, T_\circ(\nabla_\circ\hat{\varphi}^m)\,\big]. \qquad (15)$$

As before, $\Delta^{nm}\sigma(p)$ determines if there is an expected change in sign between the n-th gradient measurement $\nabla\hat{\varphi}^n(p)$ of p, and the m-th measurement $\nabla\hat{\varphi}^m(q)$ of $q = T_\circ(p)$.

Some components are more likely to belong together than others. Thus, let the weight μ_{pq}^{nm} (defined below) be proportional to the probability of $\nabla\hat{\varphi}^n(p)$ and $\nabla\hat{\varphi}^m(q)$ belonging to the same phase component θ^k. We transform Eq. (15) into a *weighted* least-squares system, that simultaneously searches for the unwrapped phase components φ^n that best adapt to *all* couplings between measurements, but giving greater weight to the most probable ones (through μ^{nm}):

$$\min_{\varphi^n} \sum_{\circ \in \{x,y\}} \sum_{\forall n} \sum_{\forall m} \big\| \mu_\circ^{nm} \{L_\circ^{nm} - f_\circ^{nm}\} \big\|^2 + \lambda \sum_{\forall n} \|\varphi^n\|^2. \qquad (16)$$

Here, $L_\circ^{nm}$ is the left-hand-side of Eq. (15), and $f_\circ^{nm}$ is its right-hand side. Furthermore, it is understood that n and m only vary over the valid indices for each pixel. For μ_{pq}^{nm}, we use the wave alignment measure proposed by Gastal and Oliveira [2] (which determines how well the local Gabor approximations, associated

(a) Original photograph (b) Our multi-channel frequency adjustment (c) Spectral remapping [2]

Fig. 6. Our multi-channel phase unwrapping leads to a frequency-adjusted image, in (b), that is faithful to the colors of the original photograph, in (a). The remapping algorithm of Gastal and Oliveira [2], in (c), reduces frequencies but is not able to correctly preserve colors. Adjustment of $\alpha = 0.2$. Original photograph © Monceau on Flickr.

with $\nabla\hat{\varphi}^n(p)$ and $\nabla\hat{\varphi}^m(q)$, match around the midpoint between p and q), scaled by a frequency-vector orientation measure.

C. Processing Color Images

Frequency adjustment should be performed in all channels of a color image, since they have correlated information [14]. Thus, while each color channel has its own phase decomposition (Eq. (3)), their phase functions are unwrapped simultaneously, in a single optimization step. Let $\mathcal{E}_\mathrm{c}$ be the energy functional from Eq. (16), written for the phase components φ^n_c of a specific color channel $\mathrm{c} \in \{\mathrm{R}, \mathrm{G}, \mathrm{B}\}$. We minimize the global energy

$$(\mathcal{E}_\mathrm{R} + \mathcal{E}_\mathrm{G} + \mathcal{E}_\mathrm{B}) + \beta\,(\mathcal{X}_\mathrm{RG} + \mathcal{X}_\mathrm{GB} + \mathcal{X}_\mathrm{RB}), \qquad (17)$$

where $\mathcal{X}_{\mathrm{c_1c_2}}$ are cross-channel phase constraints, between two channels c_1 and c_2, and $\beta = 10^{-3}$ (chosen empirically). These new constraints seek similar phase solutions $\varphi^n_{\mathrm{c}_1}(p)$ and $\varphi^m_{\mathrm{c}_2}(p)$, for a pixel p, if the measured instantanous frequencies $\nabla\hat{\varphi}^n_{\mathrm{c}_1}(p)$ and $\nabla\hat{\varphi}^m_{\mathrm{c}_2}(p)$ are similar, between channels c_1 and c_2. The indices $n = 1 \ldots N_{\mathrm{c}_1}(p)$ and $m = 1 \ldots N_{\mathrm{c}_2}(p)$ range over all detected phase components for pixel p at, respectively, color channels c_1 and c_2.

The cross-channel constraints $\mathcal{X}_{\mathrm{c_1c_2}}$ are expressed as

$$\mathcal{X}_{\mathrm{c_1c_2}} = \sum_{n=1}^{N_{\mathrm{c}_1}} \sum_{m=1}^{N_{\mathrm{c}_2}} \eta^{nm}_{\mathrm{c_1c_2}}\,\left(\varphi^n_{\mathrm{c}_1} - \Delta^{nm}_{\mathrm{c_1c_2}}\sigma\,\varphi^m_{\mathrm{c}_2}\right)^2, \qquad (18)$$

where $\Delta^{nm}_{\mathrm{c_1c_2}}\sigma(p)$ determines if there is an expected change in sign between the gradient vectors $\nabla\hat{\varphi}^n_{\mathrm{c}_1}(p)$ and $\nabla\hat{\varphi}^m_{\mathrm{c}_2}(p)$. Similarly, the weight $\eta^{nm}_{\mathrm{c_1c_2}}(p)$ measures the similarity of such vectors, according to their orientations and magnitudes.

As shown in Figure 6, when compared to the color-processing algorithm of Gastal and Oliveira [2], our multi-channel optimization leads to fewer color distortions when used for spatial frequency adjustments.

V. Implementation Details

Eq. (16) or (17) is solved once in a pre-processing step and the phase values $\varphi^n(p)$ are stored for subsequent real-time frequency adjustment. The resulting linear system is sparse and positive definite, which we solve using a QR factorization from SuiteSparse [15]. Frequency adjustment by a factor $\alpha > 0$ is obtained simply by replacing θ^n with $\alpha\varphi^n$ in Eq. (3). The resulting frequency-adjusted image is then reconstructed as

$$h_\alpha = s_\alpha + d, \quad \text{where} \quad s_\alpha(p) = \sum_{n=1}^{N(p)} B^n(p)\cos(\alpha\varphi^n(p)). \qquad (19)$$

In practice, it is best to compute $s_\alpha(p)$ using a neighborhood Ω_p around p. This is achieved through the use of a weighted reconstruction window w, centered at p and covering Ω_p. This guarantees a smooth reconstruction for s_α, even in the presence of noisy measurements or missing phase values. The corresponding windowed-reconstruction equation becomes

$$s_\alpha(p) = \sum_{q\in\Omega_p} \sum_{n=1}^{N(q)} w(p-q)\,B^n(q)\cos\left(\alpha\{\varphi^n(q) + \nabla\hat{\varphi}^n(q)\cdot(p-q)\}\right). \qquad (20)$$

The expression $\varphi^n(q) + \nabla\hat{\varphi}^n(q)\cdot(p-q)$ is a first order (linear) extrapolation of the phase $\varphi^n(q)$ from q to p, where $p\cdot q$ is the dot product between the points p and q. In Eq. (20), w is given by the square of an L^2-normalized Gaussian, with the same standard deviation as the Gabor atoms [2]. A complexity analysis of our algorithm, its comparison against Spectral Remapping, and a discussion on limitations can be found in the full text.

A. Processing Videos and 3-D Volumes

While one could consider independently applying our method to the individual frames, this introduces visual artifacts due to discontinuities in the recovered phases for adjacent frames. A more robust solution processes the whole 3-D (x, y, t) video volume at once. For the phase unwrapping pre-processing step, this is done by adding a new set of linear constraints relating to the third dimension: that is, simply by considering $\circ \in \{x, y, t\}$ in Eqs. (12) and (16) and all associated equations. The resulting 3-D phase functions $\varphi^n(x, y, t)$ are stored and later employed for real-time frequency adjustment by combining the frequency-adjusted components $\cos(\alpha\,\varphi^n(x, y, t))$ according to Eq. (20) (windowed reconstruction with a 3-D neighborhood Ω_p). As in the case for images, this process is completely independent per pixel and can be computed in parallel on a GPU. Furthermore, this can be done during video playback where only the current frame has to be reconstructed. The same considerations apply for processing 3-D volumes in general (3-D scalar fields).

VI. Conclusions & Publications

In this text we highlighted the contributions from the M.Sc. dissertation "Real-Time Frequency Adjustment of Images and Videos". These contributions were presented at the Eurographics 2021 conference and published as an article in the Computer Graphics Forum journal (Qualis A1) [16].

We proposed the first real-time technique for manipulation of local structured high-frequency content of images and videos. Our work also contributes a phase-unwrapping technique to recover the phase of real waves, and a technique for processing color images for use with frequency adjustments. By performing frequency adjustment with real-time feedback, our technique expands the range of artistic and technical possibilities for image and video processing.

ACKNOWLEDGEMENTS

We would like to thank Prof. Manuel M. Oliveira for the insightful discussions and for his contributions to this work. This work was sponsored by CNPq-Brazil (fellowships and grants 131731/2019-0, 436932/2018-0, 131102/2018-4, 312975/2018-0, 423673/2016-5), and financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

REFERENCES

[1] S. G. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed. Elsevier, 2009.

[2] E. S. L. Gastal and M. M. Oliveira, "Spectral remapping for image downscaling," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 145:1–145:16, 2017.

[3] D. Nehab and H. Hoppe, "A fresh look at generalized sampling," *Foundations and Trends in C.G. and Vision*, vol. 8, no. 1, pp. 1–84, 2014.

[4] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*. Pearson Higher Education, 2010.

[5] D. Childers, D. Skinner, and R. Kemerait, "The cepstrum: A guide to processing," *Proc. of the IEEE*, vol. 65, no. 10, pp. 1428–1443, 1977.

[6] R. M. Goldstein, H. A. Zebker, and C. L. Werner, "Satellite radar interferometry: Two-dimensional phase unwrapping," *Radio Science*, vol. 23, no. 4, pp. 713–720, Jul. 1988.

[7] N. Delprat, B. Escudie, P. Guillemain, R. Kronland-Martinet, P. Tchamitchian, and B. Torresani, "Asymptotic wavelet and Gabor analysis: extraction of instantaneous frequencies," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 644–664, Mar. 1992.

[8] J. M. Lilly and S. C. Olhede, "On the Analytic Wavelet Transform," *IEEE Trans. on Information Theory*, vol. 56, no. 8, pp. 4135–4156, Aug. 2010.

[9] P. Guillemain and R. Kronland-Martinet, "Characterization of acoustic signals through continuous linear time-frequency representations," *Proceedings of the IEEE*, vol. 84, no. 4, pp. 561–585, Apr. 1996.

[10] D. C. Ghiglia and M. D. Pritt, *Two-dimensional phase unwrapping: theory, algorithms, and software*. Wiley, Apr. 1998.

[11] L. Sacht and D. Nehab, "Optimized Quasi-Interpolators for Image Reconstruction," *IEEE TIP*, vol. 24, no. 12, pp. 5249–5259, 2015.

[12] L. N. Mertz, "Speckle imaging, photon by photon," *Applied Optics*, vol. 18, no. 5, pp. 611–614, Mar. 1979.

[13] J. Solomon, *Numerical algorithms: methods for computer vision, machine learning, and graphics*. CRC Press, Taylor & Francis, 2015.

[14] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 34–41, Jul. 2001.

[15] T. A. Davis, "Algorithm 915, SuiteSparseQR: Multifrontal Multithreaded Rank-revealing Sparse QR Factorization," vol. 38, no. 1, pp. 8:1–8:22, 2011.

[16] R. L. Germano, M. M. Oliveira, and E. S. L. Gastal, "Real-time frequency adjustment of images and videos," *Computer Graphics Forum*, vol. 40, no. 2, pp. 23–37, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.142612