Image stitching by means of adaptive normalization

Oleg V. Michailovich; University of Waterloo; Waterloo, Ontario/Canada

Abstract

Image stitching – the process of amalgamation of separate image fragments to form a complete representation of the entire scene – might become quite a challenging problem in the presence of non-additive noises and/or brightness variability artifacts. An additional degree of complication may further be inflicted in situations when one is dealing with large size data, as it is usually the case in tiling microscopy. To overcome such difficulties, a novel approach to the problem of image stitching is proposed here. In the heart of the proposed solution is Wallis filtering, which is a standard tool of image processing used for adaptive contrast adjustment and local image normalization. More importantly, Wallis filtering allows representing a given image in terms of its normalized version and associated local statistics. Subsequently, we show that stitching the output of the Wallis filter is a much simpler and much more stable task as compared to stitching the images in their original domain. The proposed method has an additional advantage of being computational efficient, which is particularly important in tiling microscopy, where a typical height/width of data images is on the order of tens of thousands of pixels.

Introduction

In imaging sciences, the process of combining together the independently acquired fragments of a digital image to form a complete representation of the entire scene is known as *image stitching* or *mosaicking* [1]. Super-resolution, photo mosaicking, video stitching, texture synthesis, and object insertion are examples of applications in which image stitching has been routinely employed [2]. The present work is a result of application oriented research as well. Specifically, it has been driven by the application of image stitching in *tiling microscopy* (TM) – a common methodology used in optical microscopy to obtain high-resolution images of large-area specimens [3]. In a typical TM setting, a given specimen is mechanically shifted through a predefined filed-of-view, which allows one to acquire a complete image of the specimen by capturing its localized fragments. Such image fragments (conventionally referred to as either *tiles* or, in the case of unidirectional shifts, *stripes*) are subsequently combined together by means of an image stitching procedure.

At its core, image stitching consists of two fundamental components, *viz.* alignment and blending. As its name suggests, the former is used to bring individual fragments of the entire image into a close correspondence with respect to each other. Blending, on the other hand, controls the image values over those regions where the contents of its fragments happen to overlap. In the case of TM, the use of ultra high precision servomotors in combination with advanced electronics allows controlling the position of specimens with a very high degree of accuracy, thereby making the problem of image alignment somewhat less critical. Image blending, on the other hand, turns out to be a much less trivial problem to solve, which makes it the main focus of this work.

The current arsenal of image blending tools is diverse. Thus, for example, in situations when input image fragments (e.g., tiles) do not exhibit any noticeable differences in their respective average brightnesses, effective blending can be achieved by means of relatively simple methods such as optimal seam algorithms [4, 5] and feathering (aka alpha blending) [3, 6]. Unfortunately, neither of the above techniques guarantees to produce *seamless* results when the average brightnesses of the input images do differ considerably, as it is often the case in TM. In this respect, better results can usually be obtained using pyramid blending [7], which applies different feathering masks to different levels of a multi-resolution decomposition of the input images. Alas, this method is still not completely immune to the brightness variability problem, which is a serious drawback considering its added computational complexity. Recently, a group of blending algorithms have been proposed based on the ideas of gradient-based image editing [8]. The methods of this group exploit numerical optimization to find an optimal solution whose gradient gives the closest possible fit to the gradients of the input images with respect to a predefined distance measure and boundary conditions [9, 10, 11]. The dependence on image gradient (rather than image values) makes such methods substantially less susceptible to the brightness variability problem. Unfortunately, virtually all methods of this family exploit an *additive* model to account for various sources of (both measurement and model) errors, which does not seem to be an accurate enough assumption to be used in TM.

One of the most critical problems considerably complicating image stitching in TM stems from the presence of brightness variability artifacts, which cause the average image brightness to vary from tile to tile. The problem appears to be particularly acute in the case of fluorescence microscopy, where it is common to adjust the orientation of the focal plane/trajectory for each tile independently so as to maximize the overall image contrast. However, due to a highly nonlinear nature of the spatial distribution of stain throughout the specimen volume, situations are frequent in which the above adjustment causes the tiles to exhibit different patterns of image brightness. As a result, there might be considerable mismatches between the values of acquired tiles in the areas of their overlap.

In view of the above said, the main goal of this work has been to develop an image stitching tool which would be as immune as possible to brightness variability artifacts. To this end, we take advantage of Wallis filtering [12, 13] which allows representing a given image in terms of its normalized version along with its local statistics of the first and second order. Subsequently, we show that stitching the output of Wallis filtering (followed by back transformation) constitutes a simpler and much more stable task as compared to stitching the images in their original domain. The proposed method has an additional advantage of being computational efficient, which is particularly important in TM, where a typical image height/width is on the order of tens of thousands

of pixels.

Image stitching

To set ground for subsequent developments, we begin with a brief outline of two particular methods of image stitching, *viz.* feathering [6] and gradient-based stitching [10]. Without any loss of generality, we restrict our discussion to processing of two grayscale images, namely *u* and *v*. Also, in what follows, both images are assumed to have already been properly aligned and thus representing two fragments/views of the same scene. Formally, let Ω_u and Ω_v denote the domains of *u* and *v*, respectively. Additionally, the images are assumed to overlap over $\Omega_w := \Omega_u \cap \Omega_v \neq \emptyset$, while the final (stitched) image *g* will be defined over the combined domain $\Omega := \Omega_u \cup \Omega_v$. (See Fig. 1 for an example of a standard domain partition.) It is also instrumental to define *extensions* of *u* and *v* on the combined domain Ω as follows

$$\tilde{u}(\mathbf{r}) = \begin{cases} u(\mathbf{r}), & \text{if } \mathbf{r} \in \Omega_u \\ 0, & \text{otherwise} \end{cases}, \quad \forall \mathbf{r} \in \Omega, \quad (1)$$

and

$$\tilde{v}(\mathbf{r}) = \begin{cases} v(\mathbf{r}), & \text{if } \mathbf{r} \in \Omega_v \\ 0, & \text{otherwise} \end{cases}, \quad \forall \mathbf{r} \in \Omega. \quad (2)$$

Note that, as opposed to *u* and *v*, the extended images are defined over the whole Ω.

DOMAIN PARTITION

WEIGHTING FUNCTION
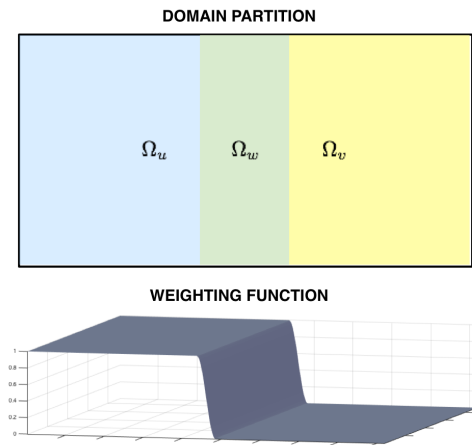
Figure 1. *Domain partition and example of a weighting function*

As our next step, we define a *weighting function* (aka a feathering mask) *W*, which has Ω as its domain of definition and takes values in the interval $[0,1]$, i.e., $0 \leq W(\mathbf{r}) \leq 1$, for all $\mathbf{r} \in \Omega$. We also require *W* to be equal to 1 over $\Omega_u \backslash \Omega_w$, while being equal to zero over $\Omega_v \backslash \Omega_w$. Finally, it is customary to work with continuous (or even differentiable) weights, which suggests that *W* has a slow transition over Ω_w, where its values descend from a maximum of 1 to a minimum of 0. An example of such function is shown at the bottom subplot of Fig. 1.

Using the above definitions of the relevant domains and the weighting function *W*, it is now straightforward to describe the method of image stitching by feathering, which consists in computing the values of *g* as given by

$$g(\mathbf{r}) = W(r)\tilde{u}(\mathbf{r}) + (1 - W(r))\tilde{v}(\mathbf{r}), \quad \forall \mathbf{r} \in \Omega. \quad (3)$$

Note that the values of *g* coincide with those of *u* and *v* over the subsets $\Omega_u \backslash \Omega_w$ and $\Omega_v \backslash \Omega_w$, respectively, whereas over the transition region Ω_w, the values of *g* represent a convex combination of the corresponding values of the input images. Due to this property, stitching by feathering could be possibly described as one of the most conservative approaches, in the sense that all the associated approximations are restricted to the transition/overlap region Ω_w exclusively. On top of that, feathering is undoubtedly one of the most computationally efficient methods, which often makes it a method of choice in applications where one deals with images of relatively large dimensions (as it is the case in digital pathology) [3].

Unfortunately, stitching by feathering fails to produce satisfactory results in situations when the mean intensities of input images differ considerably over Ω_w. In such cases, more sophisticated stitching methods need to be employed. Among such methods are gradient-based techniques [9, 10, 11]. Within this group of methods, a typical way to recover a stitched image *g* is by means of solving the following minimization problem

$$\min_g \Big\{ \int_\Omega W(\mathbf{r}) \|\nabla g(\mathbf{r}) - \nabla \tilde{u}(\mathbf{r})\|_p^p d\mathbf{r} +$$
$$+ \int_\Omega (1 - W(\mathbf{r})) \|\nabla g(\mathbf{r}) - \nabla \tilde{v}(\mathbf{r})\|_p^p d\mathbf{r} \Big\}, \quad (4)$$

with $\|\cdot\|_p$ denoting the *p*-norm of corresponding vectors. The above formulation requires the gradient of an optimal *g* to be as close as possible to the gradients of *u* and *v* (over their respective domains of definition), with the nature of this "closeness" being controlled by the value of *p*, typically set to be equal to either 1 or 2 [10]. It deserves noting that some authors suggest to concurrently minimize over both *g* and *W*, thereby allowing the weighting function *W* to be chosen adaptively [11]. Unfortunately, the resulting minimization problem ceases to be convex, which is a serious drawback that is further exacerbated by a substantial increase in computational complexity associated with the necessity to control the smoothness, boundedness, and monotonicity of *W*. Consequently, despite the improved generality of its formulation, concurrently minimizing over *g* and *W* comes with weaker guarantees of global convergence and (often prohibitive) computational burden, which altogether reduces the value of the method for applications dealing with large-size images.

For the above reasons, in this work, we adopt the more conventional formulation of (4), while setting $p = 2$. Although less robust in comparisom to the case of $p = 1$, the above choice of the norm offers considerable practical advantages (while yielding stable and useful solutions as demonstrated later in this paper). In particular, Theorem 2 in [10] shows that, under rather general conditions, setting $p = 2$ allows one to reduce (4) to an equivalent minimization problem of the form

$$\min_g \Big\{ \int_\Omega \|\nabla g(\mathbf{r}) - F\|_2^2 d\mathbf{r} \Big\}, \quad (5)$$

where

$$F(\mathbf{r}) = W(\mathbf{r})\nabla \tilde{u}(\mathbf{r}) + (1 - W(\mathbf{r}))\nabla \tilde{v}(\mathbf{r}), \quad \forall \mathbf{r} \in \Omega. \quad (6)$$

In other words, the optimal *g* minimizes $\mathscr{L}_2$-distance between its gradient and the vector field *F*, which is nothing else but the result of stitching of the gradients of *u* and *v*.

The problem in (5) is a classical (convex) optimization problem, whose associated Euler-Lagrange optimality condition leads to the Poisson equation of the form

$$\nabla^2 g = \rho \text{ in } \Omega,$$
$$\frac{\partial g}{\partial \mathbf{n}} = 0 \text{ on } \partial\Omega, \tag{7}$$

with $\rho := \text{div}(F)$ and $\partial g/\partial \mathbf{n}$ denoting the directional derivative of g along the normal $\mathbf{n}$ to the boundary $\partial\Omega$ of Ω. Note that the Neumann boundary conditions used in (7) correspond to symmetric (aka reflective) boundary conditions on g, which are common throughout imaging sciences [14]. Moreover, under the assumption on domain Ω to be rectangular, the problem (7) admits a closed-form solution, which is obtained through inverse filtering in the domain of discrete cosine transform (DCT) [15]. Specifically, let Ω be a $N \times M$ uniform rectangular lattice, and let C be the *inverse filter* defined according to

$$C[k,l] = \begin{cases} 0, & \text{if } k = l = 0, \\ [2\cos(\pi k/N) + 2\cos(\pi l/M) - 4]^{-1}, & \text{otherwise,} \end{cases} \tag{8}$$

where $k = 0, 1, \ldots, N-1$ and $l = 0, 1, \ldots, M-1$. Then, denoting by DCT and DCT^{-1} the operators of direct and inverse DCT, respectively, the unique solution to (7) is given by

$$f = \text{DCT}^{-1}\{\text{DCT}\{\rho\} \cdot C\}, \tag{9}$$

where the dot stands for point-wise matrix multiplication. It goes without saying that the relatively high computational efficiency of DCT renders the gradient-based stitching particularly attractive in a variety of practical settings, and especially when large-scale images are involved. For the convenience of referencing, in what follows, the above method will be referred to as *Poisson stitching*, with its computational structure outlined in Algorithm 1 below.

Algorithm 1 Poisson stitching

procedure POISSON(u, v, W, C)
$\tilde{u} \leftarrow$ extend u using (1)
$\tilde{v} \leftarrow$ extend v using (2)
$F \leftarrow W \cdot \nabla\tilde{u} + (1-W) \cdot \nabla\tilde{v}(\mathbf{r})$
$g \leftarrow \text{DCT}^{-1}\{\text{DCT}\{\text{div}(F)\} \cdot C\}$
return g

Despite its solid theoretical foundation, Poisson stitching still has a number of limitations, the least critical of which stems from the property of the inverse filter to have a zero response at $k = l = 0$. This property suggests that the mean value of resulting g is always equal to zero, which is a rather mild deficiency that is easy to rectify by offsetting g by a constant value so as to make the result saturate a desired dynamic range. A more series limitation of the method, however, is related to its underlying assumptions. In particular, from the viewpoint of statistical estimation, the form of the cost functional in (5) suggests that F is considered to be a *noisy* version of ∇g, with the noise being of an *additive* and *Gaussian* nature. Unfortunately, the above assumption does not seem to be well-justified in TM, where actual noises are, in fact, Poissonian. It would neither be right to apply the additive & Gaussian

model to describe the brightness inhomogeneity artifacts caused by nonuniform spatial distribution of fluorescence across a given specimen.

The inadequacy of statistical modelling implied by (5) with respect to the nature of actual imaging artifacts/noises inherent in TM severely affects the effectiveness and reliability of Poisson stitching. The problem becomes particularly acute in the case of large-scale images. Indeed, it is easy to see that, in a close vicinity of zero frequency (aka DC), the gain of the inverse filter C is on the order of $-\max\{N^2, M^2\}/\pi^2$, which reaches astronomical values when N and/or M approach the size of tens of thousands. On such conditions, even the slightest inaccuracies induced by the "model-measurement mismatch" tend to destabilize the solution, thereby rendering it completely unreliable. As futile as it sounds, however, the situation can be salvaged by means of a simple procedure detailed in the next section.

Proposed method

The principal idea of the proposed approach is based on the notion of Wallis filtering, which is a standard image processing tool used for adaptive contrast adjustment [12, 13]. The filter exploits a fundamental result in statistics stating that if x is a random variable obeying a Gaussian distribution with mean μ and standard deviation σ, i.e., $x \sim \mathcal{N}(\mu, \sigma)$, then its *normalized* version $x' := (x - \mu)/\sigma$ is distributed according to $\mathcal{N}(0, 1)$. In application to imaging, the filter operates with the *local* means and variances of a given image f. To define these statistics, let $N(\mathbf{r})$ denote a local neighbourhood of some arbitrary (yet fixed) pixel $\mathbf{r}$. Also, let $\{w(\mathbf{r}')\}_{\mathbf{r}' \in N(\mathbf{r})}$ be a set of shift-invariant weights that satisfy $\sum_{\mathbf{r}' \in N(\mathbf{r})} w(\mathbf{r}') = 1$, for all $\mathbf{r}$. Then, the local means of f can be computed according to

$$f_m(\mathbf{r}) = \sum_{\mathbf{r}' \in N(\mathbf{r})} w(\mathbf{r}') f(\mathbf{r}'), \tag{10}$$

for all $\mathbf{r} \in \Omega$. Similarly, the local variances of f can be computed as given by

$$f_s(\mathbf{r}) = \sum_{\mathbf{r}' \in N(\mathbf{r})} w(\mathbf{r}')\big(f(\mathbf{r}') - f_m(\mathbf{r})\big)^2. \tag{11}$$

Subsequently, using f_m and f_s (which can be viewed as two images of the same size as f), one can computed a normalized version f_n of the original image according to

$$f_n(\mathbf{r}) = (f(\mathbf{r}) - f_m(\mathbf{r}))/\sqrt{f_s(\mathbf{r})}, \tag{12}$$

for all $\mathbf{r}$. Equivalently, we could write $f_n = (f - f_m)/\sqrt{f_s}$, with the slash denoting point-wise division and $\sqrt{f_s}$ being an $N \times M$ array obtained by element-wise application of square root to the respective elements of f_s.

Algorithm 2 Wallis filtering

procedure WALLIS(f, h)
$f_m \leftarrow f * h$
$f_s \leftarrow (f \cdot f) * h - f_m \cdot f_m$
$f_n \leftarrow (f - f_m)/\sqrt{f_s}$
return f_n, f_m, f_s

Note that the cost of computing f_n is dominated by the cost of computing f_m and f_s, which amounts to two convolutions. To

see that, we assume $N(\mathbf{r})$ to be a rectangular neighbourhood of size $L_N \times L_M$ and choose h to be a convolution kernel defined by $h = (L_N L_M)^{-1} \mathbf{1}_{L_N} \mathbf{1}_{L_M}^T$, with $\mathbf{1}_{L_N}$ (resp. $\mathbf{1}_{L_M}$) being a column vector of ones of length L_N (resp. L_M). Then, Wallis filtering can be implemented using Algorithms 2 specified above. Here, the symbols $\cdot$ and $/$ denote point-wise multiplication and division, while $*$ stands for the operation of 2-D convolution. Note that, due to the separability of h, the 2-D convolution can be reduced to a series of 1-D convolutions along the row and column directions using kernels $h_N = (L_N)^{-1} \mathbf{1}_{L_N}$ and $h_M = (L_M)^{-1} \mathbf{1}_{L_M}$, respectively. Moreover, often better estimation results can be obtained when the "box" kernels are replaced with "bell-shaped" ones. Thus, the results reported in this paper have been obtained using Gaussian filters of standard deviation σ_h.

Before moving forward, it is worthwhile noting that, in practice, it is common to further modify the results of Wallis filtering according to $f_n \cdot \alpha + \beta$ (with $\alpha, \beta \geq 0$ being either scalars or $N \times M$ arrays) with the obvious substitution $\alpha = \sqrt{f_s}$ and $\beta = f_m$ restoring the original image f. It is also instructive to observe that while f_m and f_s contain information about local variations of the brightness and contrast of f across its domain of definition, the normalized image f_n contains information on structural variations and finer details (e.g., edges, textures, etc.) In other words, *brightness variability artifacts present in the original image f are likely to persist in f_m and f_s, whereas affecting f_n to a much lesser degree*. The latter observation lies in the heart of the proposed stitching algorithm and is further illustrated in Fig. 2. The upper subplots of the figure show two images (u and v) acquired by means of tiling fluorescence microscopy (TFM). The images are required to be stitched in the horizontal direction, which seems to be a rather non-trivial task in view of the obvious differences between the images' brightness. Indeed, in this case, stitching by feathering produces a noticeable seam as can be seen in the leftmost subplot at the bottom of Fig. 2. However, when the same stitching routine is applied to the normalized versions of u and v, the output no longer exhibits any noticeable artifacts, as shown in the rightmost subplot of the same figure.

Although encouraging, the above result is obviously premature to be used to devise a workable solution. The main reason for this is that stitching of normalized images produces results which are devoid of local brightness and contrast information. On the other hand, this information is still contained in the mean and variance images, which therefore need to be properly stitched as well. To this end, we propose the following stitching strategy:

1. First, the input images u and v are subjected to Wallis filtering, resulting in their respective triples of outputs, namely $\{u_n, u_m, u_s\}$ and $\{v_n, v_m, v_s\}$.
2. Due to the "flattened" nature of the normalized images u_n and v_n, their stitching can be performed by means of a standard feathering procedure, as defined by (3). The resulting image is denoted by g_n.
3. The mean images u_m and v_m contain information about the spatial variability of locally averaged image brightness, which makes them susceptible to brightness inhomogeneity artifacts. As a result, stitching the images by means of feathering would likely be prone to errors, and hence needs to be avoided. Instead, we proposed to use Poisson stitching, as detailed in Algorithm 1. The resulting image is denoted by

Image u   **Image v**

Feathering (original)   **Feathering (normalized)**
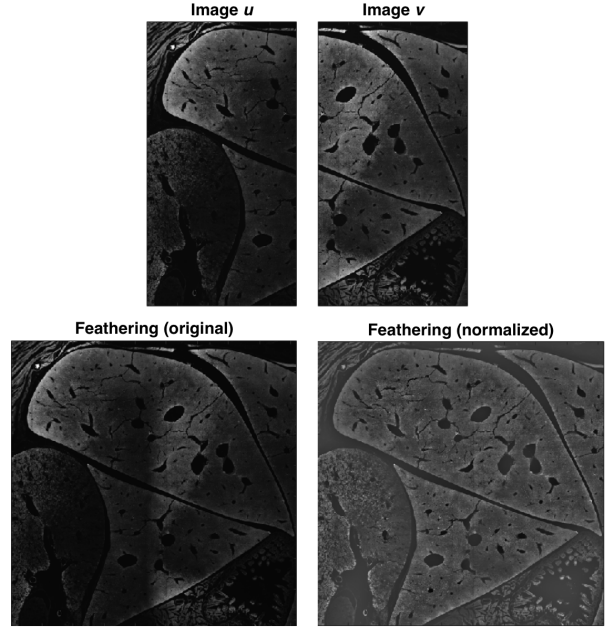
Figure 2. Stitching by feathering in the original (left) and normalized (right) domains.

g_m.

4. In a similar manner, the variance images u_s and v_s are stitched by means of the Poisson stitching algorithm as well. The resulting image is denoted by g_s.
5. Finally, the output image g is computed by combining the normalized image g_n with its associated mean and variance images according to

$$g = g_n \cdot \sqrt{g_s + \alpha} + (g_m + \beta). \tag{13}$$

Note that in the expression above, the constants $\alpha > 0$ and $\beta > 0$ are used to offset the images g_s and g_m, respectively, which otherwise would have had zero mean values. In practice, these constants are set so as to allow image g to saturate a predefined dynamic range.

The approach described above is summarized in the form of a pseudocode shown in Algorithm 3 below. For the convenience of referencing, in what follows, the proposed method will be referred to as Wallis-Poisson stitching (WPS).

Algorithm 3 Wallis-Poisson stitching
---
procedure $\text{WPS}(u, v, h, W, C)$
  $[u_n, u_m, u_s] \leftarrow \text{WALLIS}(u, h)$
  $[v_n, v_m, v_s] \leftarrow \text{WALLIS}(v, h)$
  $\tilde{u}_n \leftarrow$ extend u_n using (1)
  $\tilde{v}_n \leftarrow$ extend v_n using (2)
  $g_n = W \cdot \tilde{u}_n + (1 - W) \cdot \tilde{v}_n$
  $g_m \leftarrow \text{POISSON}(u_m, v_m, W, C)$
  $g_s \leftarrow \text{POISSON}(u_s, v_s, W, C)$
  $g \leftarrow g_n \cdot \sqrt{g_s + \alpha} + (g_m + \beta)$
return g

Before moving to demonstration of some experimental re-sults, a few important comments need to be made to justify the application of Poisson stitching to the mean and variance images. The main question here, of course, is why this method is promised to work better in comparison to directly stitching the original u and v. To address this question we first note that each value of u_m (resp. v_m) is obtained as a linear combination of the values of u (resp. v). Consequently, the central limit theorem (CLT) suggests that, although the values of u (resp. v) are affected by Poisson noise, the noise contaminating the values of u_m (resp. v_m) will tend to be more Gaussian. Although not as straightforward as in the case of the mean images, a similar argument can be applied to the variance images u_s and v_s as well. It is, therefore, reason-able to conclude that u_m and v_m as well as u_s and v_s represent "gaussianized" versions of u and v, which places them in a much better agreement with the statistical interpretation of (5). Addi-tionally, a quick look at Algorithm 2 reveals that to compute u_m and v_m as well as u_s and v_s, the input images u and v are sub-jected to *low-pass filtering* through convolution with a Gaussian kernel h. As a result, the mean and variance images turn out to be substantially oversampled with respect to their corresponding bandwidths. This fact suggests the possibility to apply Poisson stitching to the *down-sampled* versions of the mean and variance images, followed by proper up-sampling. Consequently, in our experiments with confocal microscopy images, we have been able to reduce the sampling rate of original images by a factor of 100 (through appropriately choosing the value of σ_h), with associated aliasing errors being below 0.5%. To demonstrate the significance of this result, let us consider a practical scenario, in which one is to scan a 20mm-by-20mm specimen at a spatial resolution of 1 μm. The resulting image will have a size of 20,000-by-20,000 pixels, which would make it rather unamenable for efficient pro-cessing, even using advanced means of scientific computing. On the other hand, a 100-fold subsampled version of the image would have a size of 200-by-200 pixels, which makes its processing vir-tually effortless. What is even more important is the fact that, as long as the original image is considered, the maximum magnitude response of its associated inverse filter in (8) is around 152 dB, while the subsampling allows it to be reduced by 80 dB to a vicin-ity of 72 dB. Consequently, the smooth nature of the mean and variance images brings about the invaluable benefit of substan-tially improved numerical stability and computational efficiency.
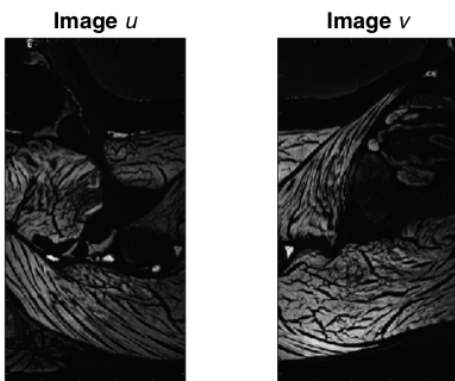

Figure 3. *Images u and v used in the experimental study.*

Results

In this section, we validate the viability and usefulness of the proposed WPS method via numerical experiments using real-life data. All the test images were acquired using TissueScope™ CF confocal fluorescence and bright-field slide scanner (Huron Digital Pathology, Waterloo, ON, Canada). The spatial resolution in both x- and y-direction was set to be equal to 1 μm. For the sake of comparative analysis, stitching by feathering and Poisson stitching have been used as reference methods. It should also be noted that, as an additional reference method, the authors tried to use the GIST1-ℓ_1 method of [10] (which amounts to solving (4) for $p = 1$). Unfortunately, for the problem at hand, the results thus obtained have been of similar quality to that of Poisson stitching, and, for this reason, their demonstration is omitted here for the reason of space.
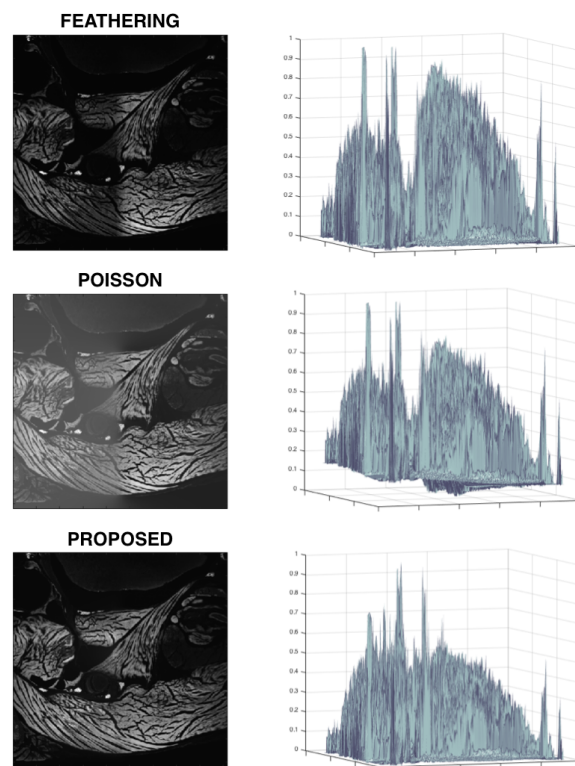

Figure 4. *Results obtained by (from top to bottom) stitching by feathering, Poisson stitching, and the proposed WPS method.*

Fig. 3 shows two fragments of image tiles (aka "strips") to be stitched along the horizontal direction. One can see that complex and inhomogeneous distribution of the stain across the specimen as well as differences in the orientations of the focal planes create significant differences in the brightness of the images. The highly non-linear nature of these brightness variability artifacts makes it extremely difficult to attain useful stitching results using standard approaches such as feathering and Poisson stitching. This point is demonstrated in Fig. 4 that shows the results of various stitch-ing methods depicted as grayscale images (left column of sub-plots) along with their corresponding surface plots (right column of subplots). As expected, the result produced by feathering suf-

fers from the presence of an unwanted (vertical) seam. The result produced by Poisson stitching, on the other hand, reveals severe distortions that manifest themselves through unnatural negative overshoots (as seen in the corresponding surface plot) and, as a result, reduced contrast. At the same time, the proposed method is capable of attaining a close to perfect performance, as shown in the bottom of Fig. 4.
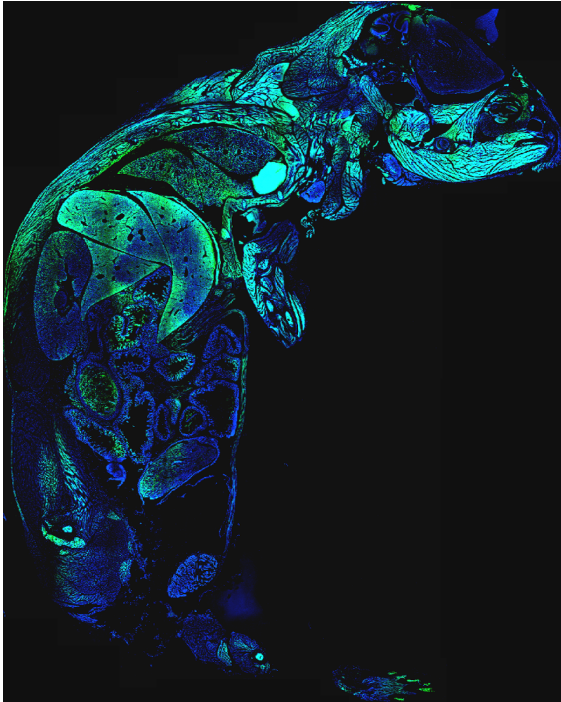
Figure 5. *Stitched image computed by the proposed algorithm.*

While the examples in Fig. 4 show various stitching results performed on a single colour channel, Fig. 5 provides an example of a full coloured reconstruction. The shown image has been formed from 10 vertical "stripes" of size 57,000-by-4,850 pixels each (with a 5 mm horizontal overlap). The full reconstruction took less than 3 mins on a standard 2.6 GHz Intel Core i7 CPU running Matlab® (version R2015a). One can see that the image exhibits neither noticeable seams nor other undesirable artifacts.

Summary

In this paper, a new method of image stitching has been proposed. The main impetus for the development of this method has come from its application to TM – an image acquisition technique in which the entire image of a relatively large specimen is formed via stitching its localized fragments. The principal advantages of the proposed method consist in its ability to perform reliably in the presence of (average) brightness variability artifacts as well as its relatively high computational efficiency. This is what makes this method optimally suited for application in tiling microscopy, where a typical image height/width is on the order of tens of thousands of pixels. It is also interesting to note that, in its philosophy, the proposed method reminds the multi-resolution approach of [7], in which the *same* stitching procedure (i.e., feathering) is applied to different levels of an image decomposition pyramid. In this work, albeit using a different decomposition mechanism (i.e., Wallis filtering), we show that additional advantages can be gained through applying *different* stitching methods to different constituents of an image decomposition.

Acknowledgments

References

[1] A. Agarwala et al., Interactive Digital Photomontage, ACM Transactions on Graphics, 23, 3, pp. 292-300 (2004).

[2] R. Szeliski, Image Alignment and Stitching: A Tutorial, Technical Report MSR-TR-2004-92, Microsoft Research (2004).

[3] S. Preibisch, S. Saalfeld and P. Tomancak, Globally Optimal Stitching of Tiled 3D Microscopic Image Acquisitions", Bioinformatics, 25, 11, pp.1463-1465 (2009).

[4] A. Efros and W. Freeman, Image Quilting for Texture Synthesis and Transfer, in Proceed. of SIGGRAPH, pp. 341-346 (2001).

[5] D. Milgram, Computer methods for creating photomosaics, IEEE Trans. Comput., 23, pp. 1113-1119 (1975).

[6] M. Uyttendaele, A, Eden and R. Szeliski, Eliminating Ghosting and Exposure Artifacts in Image Mosaics, in Proceed. of CVPR, pp. 509-516 (2001).

[7] P. J. Burt and E. H. Adelson, A Multiresolution Spline with Application to Image Mosaics, ACM Trans. Graphics, 2, 4, pp. 217-236 (1983).

[8] P. Perez, M. Gangnet and A. Blake, Poisson Image Editing, ACM Trans. Graph, 22, pp. 313-318 (2003).

[9] R. Szeliski, M. Uyttendaele and D. Steedly, Fast Poisson Blending Using Multi-Splines, in Proceed. of IEEE ICCP (2011).

[10] A. Levin, A. Zomet, S. Peleg and Y. Weiss, Seamless Image Stitching in the Gradient Domain, in Proceed. of ECCV, 3024, pp. 377-389 (2004).

[11] M. K. Ng and W. Wang, A Variational Approach for Image Stitching II: Using Image Gradients, SIAM J. Imaging Sci., 6, 3, pp. 1345-1366 (2013).

[12] K. F. Wallis, Seasonal Adjustment and Relations Between Variables, J. of the American Statistical Association, 69, 345, pp. 18-31 (1976).

[13] R. Gonzalez and R. Woods, Digital Image Processing, 3rd Edition, Prentice-Hall, Inc., 2006.

[14] T. F. Chan and J. Shen, Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods, SIAM, 2005.

[15] G. Strang, The Discrete Cosine Transform, SIAM Review, 41, 1, pp. 135-147 (1999).

Author Biography

Oleg Michailovich holds a double MASc degree in electrical and biomedical engineering from Saratov State University (1994) and Technion - IIT (2000), respectively, as well as a PhD in biomedical engineering from the Technion - IIT (2003). Currently, he is an associate professor at the Department of Electrical and Computer Engineering at the University of Waterloo (Waterloo, ON). His research focuses on inverse problems in signal and image processing as well as related computational methods. He is a member of IEEE and presently serves as an Associate Editor of IEEE Transactions on Image Processing.