Seamless Image Stitching by Minimizing False Edges

Assaf Zomet, Anat Levin, Shmuel Peleg, and Yair Weiss

Abstract—**Various applications such as mosaicing and object insertion require stitching of image parts. The stitching quality is measured visually by the similarity of the stitched image to each of the input images, and by the visibility of the seam between the stitched images. In order to define and get the best possible stitching, we introduce several formal cost functions for the evaluation of the stitching quality. In these cost functions the similarity to the input images and the visibility of the seam are defined in the gradient domain, minimizing the disturbing edges along the seam. A good image stitching will optimize these cost functions, overcoming both photometric inconsistencies and geometric misalignments between the stitched images. We study the cost functions and compare their performance for different scenarios both theoretically and practically. Our approach is demonstrated in various applications including generation of panoramic images, object blending and removal of compression artifacts. Comparisons with existing methods show the benefits of optimizing the measures in the gradient domain.**

I. INTRODUCTION

IMAGE stitching is a common practice in the generation of panoramic images and applications such as object insertion, object removal [1] and super resolution [2]. An example of image stitching is shown in Fig. 1. Two images I_1, I_2 capture different portions of the same scene, with an overlap region viewed in both images. The images should be stitched to generate a mosaic image I. A simple pasting of a left region from I_1 and a right region from I_2 produces visible artificial edges in the seam between the images, due to differences in camera gain, scene illumination or geometrical misalignments.

The aim of a stitching algorithm is to produce a visually plausible mosaic with two desirable properties. First, the mosaic should be as *similar* as possible to the input images, both geometrically and photometrically. Second, the seam between the stitched images should be *invisible*. While these requirements are widely acceptable for visual examination of a stitching result, their definition as quality criteria was either limited or implicit in previous approaches.

In this work, we present several cost functions for these requirements, and define the mosaic image as their optimum. The stitching quality in the seam region is measured in the gradient domain. The mosaic image should contain a minimal amount of seam artifacts, i.e. a seam should not introduce a new edge that does not appear in either I_1 or I_2. As image dissimilarity, the gradients of the mosaic image I are compared with the gradients of I_1, I_2. This reduces the effects caused by global inconsistencies between the stitched images. We call our framework GIST: gradient-domain image stitching.

We demonstrate this approach in several applications, including panoramic mosaicing, object blending, and removal of compression artifacts. Analytical and experimental comparisons of our approach to existing methods show the benefits in working in the gradient domain and in directly minimizing gradient artifacts.

A. Related Work

There are two main approaches to image stitching in the literature, assuming that the images have already been aligned. Optimal seam algorithms [3]–[6] search for a curve in the overlap region on which the differences between I_1, I_2 are minimal. Then, each image is copied to the corresponding side of the seam. In case the difference between I_1, I_2 on the curve is zero, no seam gradients are produced in the mosaic image I. However, the seam is visible when there is no such curve, for example, when there are globally smooth intensity differences between the images. This is illustrated on the first row of Fig. 2. In addition, optimal seam methods are less appropriate when thin strips are taken from the input images, as in the case of manifold mosaicing [7].

The second approach minimizes seam artifacts by smoothing the transition between the images. In Feathering [8] or alpha blending, the mosaic image I is a weighted combination of the input images I_1, I_2. The weighting coefficients (alpha mask) vary as a function of the distance from the seam. In pyramid blending [9] different frequency bands are combined with different alpha masks. Lower frequencies are mixed over a wide region, and fine details are mixed in a narrow region. This produces gradual transition in lower frequencies, while reducing edge duplications in textured regions. Therefore the result mosaic is sharp, while eliminating artificial edges due to differences in lower frequencies. A related approach was suggested in [10], where a smooth function was added to the input images to force a consistency between the images in the seam curve. In case there are misalignments between the images [8], these methods leave artifacts in the mosaic such as double edges, as shown in Figs. 2 and 3.

In our approach, we compute the mosaic image I by an optimization process that uses image gradients. Computation in the gradient domain was recently used in compression of dynamic range [11], image inpainting [12], and separation of images to layers [13]–[16]. There are two closely related works that used image gradients for stitching. Perez *et al.* [1] proposed a framework for image editing in the gradient domain. One application

A. Zomet was with the School of Computer Science and Engineering, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel. He is now with HumanEyes Technologies, Jerusalem, Israel (e-mail: zomet@humaneyes.com).

A. Levin, S. Peleg, and Y. Weiss are with the School of Computer Science and Engineering, The Hebrew University of Jerusalem 91904, Jerusalem, Israel (e-mail: alevin@cs.huji.ac.il; peleg@cs.huji.ac.il; yweiss@cs.huji.ac.il).
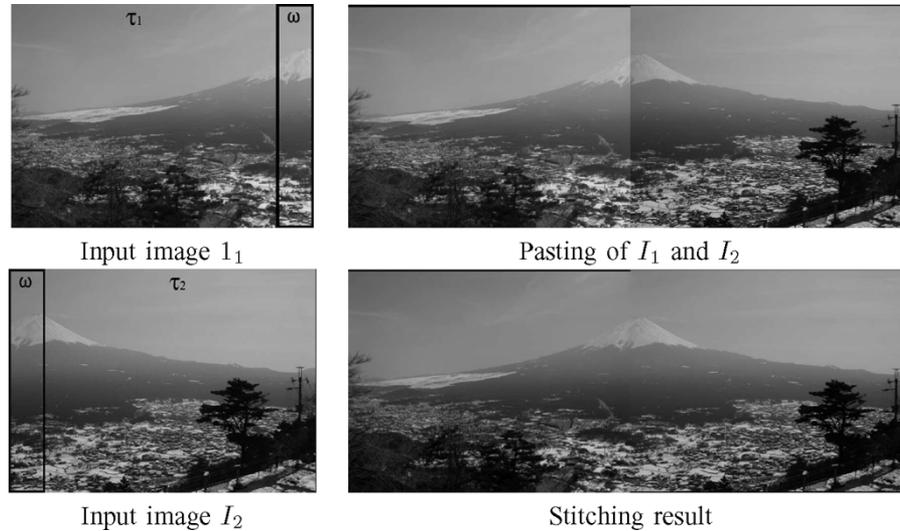
Fig. 1.   Image stitching. On the left are the input images. ω is the overlap region. Top right: Simple pasting of the input images. Bottom right: Result of the GIST1 algorithm.

Fig. 2.   Comparing stitching methods with various sources for inconsistencies between the input images. The left side of I_1 is stitched to right side of I_2. First row: Optimal seam methods produce a seam artifact in case of photometric inconsistencies between the images. Second row: Feathering and pyramid blending produce double edges in case of horizontal misalignments. Third row: In case there is a vertical misalignments, the stitching is less visible with Feathering and GIST.
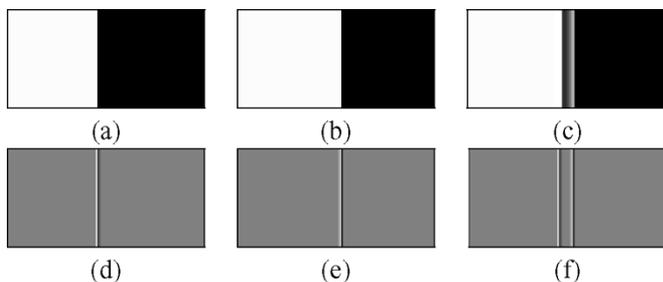

Fig. 3.   Misalignment problem in image stitching. One of the most popular methods to stitch images is pyramid blending [9], in which each frequency band is stitched separately: (a) and (b) show two misaligned input images, taken from the second row of Fig. 2; (d) and (e) show the corresponding bands of highest frequencies (top level of Laplacian pyramid); (f) shows the stitched bands and (c) shows the reconstructed image. Artifacts due to the misalignment are easily noted.

is object insertion, where an object is cut from an image, and inserted to a new background image. The insertion is done by optimizing over the gradients of the inserted object, with the boundary determined by the background image. Agarwala *et al.*

[6], in a work concurrent to ours, employ a two-stage approach. First, optimal seams are found across the multiple input images. Then, rather than composing the images, the authors compose the gradient images and derive the final mosaic using optimization. We compare these works [1], [6] to ours in Section IV.

II. GIST: IMAGE STITCHING IN THE GRADIENT DOMAIN

We describe two approaches to image stitching in the gradient domain. Section II-A describes GIST1, where the mosaic image is inferred directly from the derivatives of the input images. Section II-B describes GIST2, a two-steps approach to image stitching. Section II-C compares the two approaches to each other, and with other methods.

A. GIST1: Optimizing a Cost Function Over Image Derivatives

The first approach, GIST1, computes the stitched image by minimizing a cost function E_p. E_p is a dissimilarity measure between the derivatives of the stitched image and the derivatives of the input images.

Consider Fig. 1. Let I_1, I_2 be the two aligned input images. Let τ_1 (τ_2 resp.) be the region viewed exclusively in image I_1 (I_2 resp.), and let ω be the overlap region., with $\tau_1 \cap \tau_2 = \tau_1 \cap \omega = \tau_2 \cap \omega = \emptyset$. Let W be a weighting mask image.

The stitching result I of GIST1 is defined as the minimum of E_p with respect to $\hat{I}$

$$E_p(\hat{I}; I_1, I_2, W) = d_p(\nabla\hat{I}, \nabla I_1, \tau_1 \cup \omega, W)$$
$$+ d_p(\nabla\hat{I}, \nabla I_2, \tau_2 \cup \omega, U - W) \quad (1)$$

where U is a uniform image, $U(\vec{q}) = 1$ for all $\vec{q}$, and $d_p(J_1, J_2, \phi, W)$ is the distance between J_1, J_2 on ϕ

$$d_p(J_1, J_2, \phi, W) = \sum_{\vec{q} \in \phi} W(\vec{q}) \|J_1(\vec{q}) - J_2(\vec{q})\|_p^p \quad (2)$$

with $\| \cdot \|_p$ denoting the ℓ_p-norm.

The dissimilarity E_p between the images is defined by a weighted distance between their derivatives (gradients). A dissimilarity in the gradient domain is invariant to the mean intensity of the image. In addition it is less sensitive to smooth global differences between the input images, e.g. due to nonuniformness in the camera photometric response and due to scene shading variations. On the overlap region ω, the cost function E_p penalizes for derivatives which are inconsistent with any of the input images. In image locations where both I_1 and I_2 have low gradients, E_p penalizes for high gradient values in the mosaic image. This property is useful in eliminating false stitching edges.

The selection of the weighting mask W is addressed in Section III. The choice of norm (parameter p) has implications on both the optimization algorithm and the mosaic image. The minimization of E_p (1) for $p \geq 1$ is convex, and hence efficient optimization algorithms can be used. Section III describes a minimization schemes for E_2 and for E_1. The influence of the choice of p on the result image is addressed in the following sections, with the introduction of alternative stitching algorithms in the gradient domain.

B. GIST2: Stitching Derivative Images

A simpler approach is to stitch the derivatives of the input images.

1) Compute the derivatives of the input images $(\partial I_1/\partial x)$, $(\partial I_1/\partial y)$, $(\partial I_2/\partial x)$, $(\partial I_2/\partial y)$.
2) Stitch the derivative images to form a field $F = (F_x, F_y)$. F_x is obtained by stitching $(\partial I_1/\partial x)$ and $(\partial I_2/\partial x)$, and F_y is obtained by stitching $(\partial I_1/\partial y)$ and $(\partial I_2/\partial y)$.
3) Find the mosaic image I whose gradients are closest to F. This is equivalent to minimizing $d_p(\nabla I, F, \pi, U)$ where π is the entire image area and U is a uniform image.

In 2), any stitching algorithm may be used. We have experimented with Feathering, pyramid blending [9] and optimal seam. For the optimal seam, we used the algorithm in [4],

finding the curve $x = f(y)$ that minimizes the sum of absolute differences in the input images. In 3), the optimization under ℓ_1, ℓ_2, is described in Section III. Unlike the GIST1 algorithm described in the previous section, we found minor differences in the result images when minimizing d_p under ℓ_1 and ℓ_2.

C. Which Method to Use?

In the previous sections, we presented several stitching methods. Since stitching results are tested visually, selecting the most appropriate method may be subject to personal taste. In the following, we study the differences between the stitching methods theoretically. In Section IV, we compare their performance on images.

The theoretical analysis is based on two propositions. In Proposition 2.1 we show that GIST1 under ℓ_1 is as good as the optimal seam methods when a perfect seam exists. Hence, the power of GIST1 under ℓ_1 to overcome geometric misalignments similarly to the optimal seam methods. The advantage of GIST1 over optimal seam methods is when there is no perfect seam, for example due to photometric inconsistencies between the input images. This was validated in the experiments.

Proposition 2.2 shows an equivalence between GIST1 under ℓ_2 and Feathering of derivatives (GIST2) under ℓ_2 (Feathering derivatives is different from Feathering the images).

Both propositions provide insight into the difference between GIST1 under ℓ_1 and under ℓ_2: Under ℓ_2, the algorithm tends to mix the derivatives and hence blur the texture in the overlap region. Under ℓ_1, the algorithm tends to behave similarly to the optimal seam methods, while reducing photometric inconsistencies.

Proposition 2.1: Let I_1, I_2 be two input images for a stitching algorithm, and assume there is a curve $x = f(y)$, such that for each $\vec{q} \in \{(f(y), y)\}$, $I_1(\vec{q}) = I_2(\vec{q})$. Let U be a uniform image, $U(x, y) = 1$ for all x, y. Then, the optimal seam solution I, defined below, is a global minimum of $E_p(I; I_1, I2, U)$ defined in (1), for any $0 < p \leq 1$

$$I = \begin{cases} I_1(x, y), & x < f(y) \\ I_2(x, y), & x \geq f(y). \end{cases}$$

Proof: Let ω be the overlap region. Let ψ_1 be the region where $x < f(y)$ and let ψ_2 be the region $x \geq f(y)$. The value of $E_p(I; I_1, I2, U)$ on the image I is

$$E_p(I; I_1, I_2, U)$$
$$= 0 + d_p(\nabla I, \nabla I_1, \omega, U) + d_p(\nabla I, \nabla I_2, \omega, U)$$
$$= \sum_{\vec{q} \in \omega \cap \psi_2} \|\nabla I(\vec{q}) - \nabla I_1(\vec{q})\|_p^p$$
$$+ \sum_{\vec{q} \in \omega \cap \psi_1} \|\nabla I(\vec{q}) - \nabla I_2(\vec{q})\|_p^p$$
$$= \sum_{\vec{q} \in \omega} \|\nabla I_2(\vec{q}) - \nabla I_1(\vec{q})\|_p^p.$$

Let $\hat{I}$ be another solution. Using the concavity of $\ell_p, 0 < p \le 1$, we show that $E_p(\hat{I}; I_1, I_2, U) \ge E_p(I; I_1, I_2, U)$

$$
\begin{aligned}
&E_p(\hat{I}; I_1, I_2, U) \\
&= d_p(\nabla \hat{I}, \nabla I_1, \omega, U) + d_p(\nabla \hat{I}, \nabla I_2, \omega, U) \\
&\quad + d_p(\hat{\nabla} I, \nabla I_1, \tau_1, U) + d_p(\nabla \hat{I}, \nabla I_2, \tau_2, U) \\
&\ge \sum_{\vec{q} \in \omega} \left(\left\| \nabla \hat{I}(\vec{q}) - \nabla I_1(\vec{q}) \right\|_p^p + \left\| \nabla \hat{I}(\vec{q}) - \nabla I_2(\vec{q}) \right\|_p^p \right) \\
&\ge \sum_{\vec{q} \in \omega} \| \nabla I_2(\vec{q}) - \nabla I_1(\vec{q}) \|_p^p \\
&= E_p(I; I_1, I_2, U).
\end{aligned}
$$

Result 1: GIST1 under ℓ_1 and the optimal seam methods give the same result when there is a consistent seam between the input images.

In case there is no consistent seam, the optimal seam method and GIST1 give different results. In Section IV, we compare their results, and show the benefit in using GIST1.

The second result shows the equivalence between GIST1 under ℓ_2 and Feathering of image derivatives under ℓ_2 (GIST2).

Proposition 2.2: Let I_1, I_2 be two input images for a stitching algorithm, and let W be a Feathering mask. Let ω, the overlap region of I_1, I_2, be the entire image (without loss of generality, as $W(\vec{q}) = 1$ for $\vec{q} \in \tau_1$, and $W = 0$ for $\vec{q} \in \tau_2$). Let I_{Gist} be the minimum of $E_2(I; I_1, I_2, W)$ defined in (1). Let F be the following field:

$$
F = W(\vec{q}) \nabla I_1(\vec{q}) + (1 - W(\vec{q})) \nabla I_2(\vec{q}).
$$

Then, I_{Gist} is the image with the closest gradient field to F under ℓ_2.

Proof: Let f_x, f_y be the derivative operators, and let $F_{xj} = I_j * f_x, F_{yj} = I_j * f_y$. Then

$$
\begin{aligned}
&E_2(I; I_1, I_2, W) \\
&= \sum_{\vec{q} \in \omega} W(\vec{q}) \left[((I * f_x)(\vec{q}) - F_{x1}(\vec{q}))^2 \right. \\
&\qquad\qquad\qquad \left. + ((I * f_y)(\vec{q}) - F_{y1}(\vec{q}))^2 \right] \\
&\quad + \sum_{\vec{q} \in \omega} (1 - W(\vec{q})) \left[((I * f_x)(\vec{q}) - F_{x2}(\vec{q}))^2 \right. \\
&\qquad\qquad\qquad\qquad \left. + ((I * f_y)(\vec{q}) - F_{y2}(\vec{q}))^2 \right].
\end{aligned}
$$

Let us define P_W as the linear operator multiplying each image location $\vec{q}$ by a weight $W(\vec{q})$, and let f_x^r, f_y^r be the adjoint operators of f_x, f_y. Taking the derivatives of $E_2(I; I_1, I_2, W)$ with respect to I and equating to 0, we get for each image location $\vec{q}$

$$
\begin{aligned}
&f_x^r * P_W(I * f_x) + f_x^r * P_{1-W}(I * f_x) \\
&\quad + f_y^r * P_W(I * f_y) + f_y^r * P_{1-W}(I * f_y) \\
&= f_x^r * P_W(F_{x1}) + f_x^r * P_{1-W}(F_{x2}) \\
&\qquad + f_y^r * P_W(F_{y1}) + f_y^r * P_{1-W}(F_{y2}).
\end{aligned}
$$

Noting that $P_{1-W} + P_W$ is the identity operator, and defining $F_1 = (F_{x1}, F_{y1})$, $F_2 = (F_{x2}, F_{y2})$, $F(\vec{q}) = (F_x(\vec{q}), F_y(\vec{q})) = W(\vec{q}) F_1(\vec{q}) + (1 - W(\vec{q})) F_2(\vec{q})$, we get

$$
\left(f_x^r * f_x + f_y^r * f_y \right) * I = f_x^r * Fx + f_y^r * Fy
$$

which is a discretization of the Poisson equation [1]. Hence, the minimum of $E_2(I; I_1, I_2)$ is obtained when I is the image with the closest derivatives to F under the ℓ_2 norm. ∎

Result 2: GIST1 under ℓ_2 is equivalent to Feathering of the gradient images followed by a solution of the Poisson equation (GIST2).

III. IMPLEMENTATION DETAILS

We have implemented a minimization scheme for (1) under ℓ_1 and under ℓ_2.

Equation (1) defines a set of linear equations in the image intensities, with the derivative filter taps as the coefficients. Similarly to [13], [14] we found that good results are obtained using forward-differencing derivative filters $1/4[1 \ -1]$. In the ℓ_1 case, the results were further enhanced by incorporating additional equations using derivative filters in multiple scales. Specifically, we added the filter corresponding to forward-differencing in the second level of a Gaussian pyramid, obtained by convolving the filter $[1 \ 0 \ -1]$ with a vertical and a horizontal Gaussian filter $1/4[1 \ 2 \ 1]$. Color images were handled by applying the algorithm to each of the color channels separately.

Since no boundary conditions were used, the solution of these equations is determined up to a uniform intensity shift. This shift can be selected in various ways. We chose to set it according to the median of the values of the input image I_1 and the median of the corresponding region in the mosaic.

Proposition 2.2 shows that the minimum of (1) under ℓ_2 can be found by solving the Poisson equation for a weighted combination of the gradient fields of the input images

$$
F = \begin{cases}
\nabla I_1(\vec{q}), & \vec{q} \in \tau_1 \\
W(\vec{q}) \nabla I_1(\vec{q}) + (1 - W(\vec{q})) \nabla I_2(\vec{q}), f, & \vec{q} \in \omega \\
\nabla I_2(\vec{q}), & \vec{q} \in \tau_2.
\end{cases}
$$

The solution can be obtained by various methods, e.g. deconvolution [13], FFT [17] or multigrid solvers [18]. The results presented in this paper were obtained by FFT.

We propose two implementations to solve the linear equations under ℓ_1. The first way uses a linear programming package (we used LOQO [19]). We define a linear program as follows [20]:

$$
\text{Min} : \sum_i \left(z_i^+ + z_i^- \right)
$$

$$
\text{Subject to} : Ax + (z^+ - z^-) = b, x \ge 0, z^+ \ge 0, z^- \ge 0.
$$

The entries in matrix A are defined by the coefficients of the derivative filters, and the vector b contains the derivatives of I_1, I_2. x, is a vectorization of the result image.

The second implementation for ℓ_1 optimization is slower, but does not require a linear programming package. ℓ_1 optimization can be performed by iteratively solving a series of weighted-
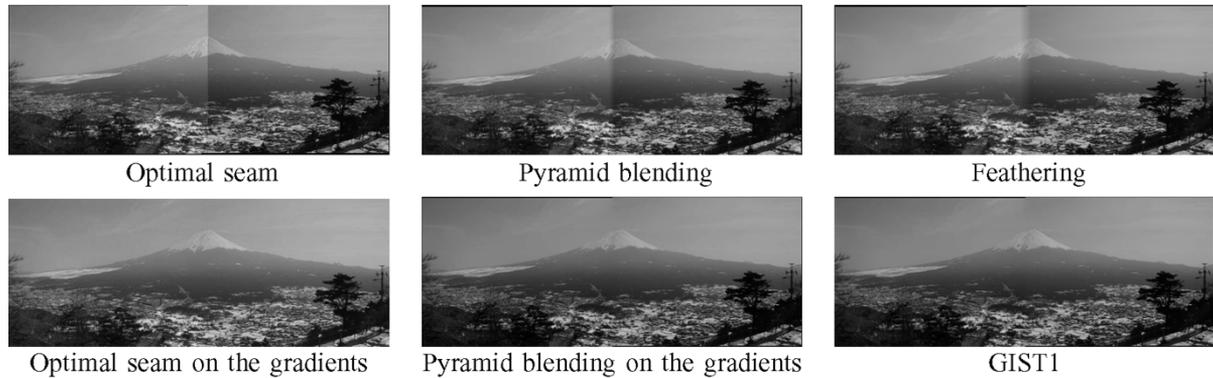
Fig. 4. Stitching in the gradient domain. The input images appear in Fig. 1 with the overlap region marked by a black rectangle. (Top panels) With the image domain methods, the stitching is observable. (Bottom panels) Gradient-domain methods overcome global inconsistencies.

least-squares optimizations [21].[1] Specifically, let $x^{(n-1)}$ be the solution estimate at iteration $n - 1$. Let e_i be the absolute value of the error in iteration $n - 1$ in equation i. In order to compute $x^{(n)}$, we define weights w_i: $w_i = (1/\max(e_i, T))$ where T is some small threshold (we used $T = 10^{-5}$). The weights w_i are ordered in a diagonal matrix O, and $x^{(n)}$ is computed by

$$x^{(n)} = (A^\top O A)^{-1} A^\top O b.$$

As an initial estimate, we used a Feathering of the input images. This algorithm performed better in terms of convergence than the algorithm proposed in [22, Sec. 3.1].

To accelerate the optimization under ℓ_1, we took the following approach: We first select an area around the center of the overlapping area, typically 40 pixels wide. We define the weighting mask over this area, and compute the mosaic on the area as described above. We then use the gradients of this computed mosaic and the gradients of the images on the corresponding sides of the area to compute the full mosaic by optimizing under ℓ_2 (solving the Poisson equation). The result of this procedure was indistinguishable in our experiments from the result of optimizing under ℓ_1 on the entire mosaic.

As for the selection of the weighting mask W, we examined two options (for vertical overlap areas)

$$W_C(x, y) = \begin{cases} 1, & x < c_x - d \\ \frac{1}{2}, & c_x - d \le x \le c_x + d \\ 0, & x > c_x + d \end{cases}$$

$$W_L(x, y) = \begin{cases} 1, & x < c_x - d \\ \frac{c_x + d - x}{2d}, & c_x - d \le x \le c_x + d \\ 0, & x > c_x + d \end{cases} \quad (3)$$

where c_x is the center of the overlap area, and d is a tunable parameter controlling the width of the transition area, that was set to be nine pixels. Under ℓ_1, we found no significant differences between the results obtained with the two masks, and we used W_C. Under ℓ_2, better results were obtained with W_L.

---

[1] In this paper, we focused on the properties of different cost function, and used a generic solver for the weighted least squares. Similarly, LOQO is a generic solver. Faster implementations can be developed in both cases that are tailored to this specific problem.
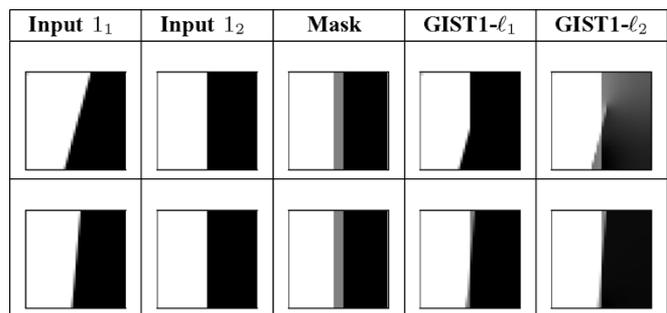
Fig. 5. Comparing optimization under ℓ_1 and under ℓ_2. With this input and mask, no mosaic is consistent with the two images. In the top example, the ℓ_2 solution has smearing artifacts that do not appear in the ℓ_1 solution. Under ℓ_1, any image whose derivatives are bounded by the derivatives of the input images is an optimal solution. Therefore, inconsistencies between the input and output images are concentrated in locations where the input images have inconsistent derivatives. In contrast, ℓ_2 tends to distribute the gradient inconsistency across the image and hence produces gradients even in areas that are uniform in both input images. The bottom example shows that, in some cases, the ℓ_1 solution and the ℓ_2 solution look similar.
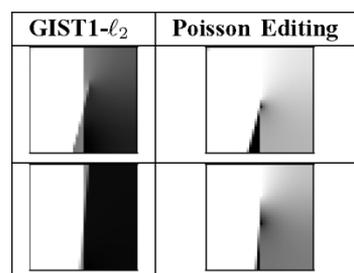
Fig. 6. Comparing optimization over a region (GIST1) to Poisson editing [1]. The input images were the same as in Fig. 5. The misalignment between the input images is destructive for Poisson editing, and may be avoided by improving the alignment. GIST1 produces cleaner results.

IV. EXPERIMENTS

In our first set of experiments, we studied the properties of the proposed stitching methods. These experiments were done mostly with synthetic images, to highlight the distinguishing features of each method. We focused on the following points.

- Gradient domain stitching versus image domain stitching.
- Optimizing under ℓ_1 versus Optimizing under ℓ_2.
- Poisson editing [1] versus our proposed method.
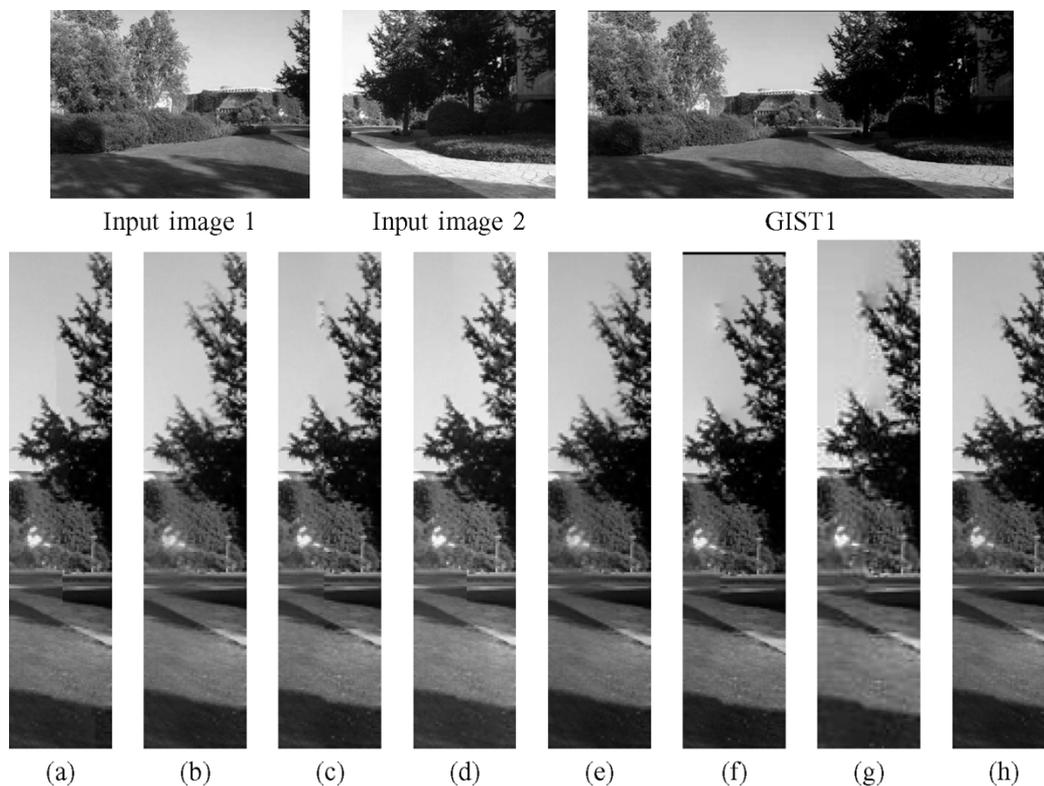
Fig. 7. Comparing various stitching methods. On the top are the input image and the result of GIST1 under ℓ_1. The images on the bottom are cropped results of various methods. (a) Optimal seam, (b) feathering, (c) pyramid blending, (d) optimal seam on the gradients, (e) feathering on the gradients, (f) pyramid blending on the gradients, (g) Poisson editing [1], and (h) GIST1—ℓ_1. The seam is visible in (a), (c), (d), and (g).

Note that the fusion stage in the work by Agarwala *et al.* [6] is similar to our approach, when optimizing under ℓ_2.

To address the first point, we compared our method to existing image stitching techniques, which work on the image intensity domain: feathering [8], pyramid blending [9], and "optimal seam" (implemented as in [4]). The experiments (Fig. 2, first row, and Fig. 4) validate the advantage in working in the gradient domain for overcoming photometric inconsistencies.

A comparison between optimization under ℓ_1 and under ℓ_2 is shown in Fig. 5. In the top example, the ℓ_2 result contains smearing artifacts that do not appear in the ℓ_1 result. This is because, under ℓ_1, any solution whose derivatives are bounded between the derivatives of both images is optimal. In other words, pixels that have zero gradients in both input images should have zero gradient also in the solution. In contrast, in the ℓ_2 solution the gradient inconsistency is being spread across a large region. This type of artifacts was also observed by Agarwala *et al.* [6]. The differences between the ℓ_1 result and the ℓ_2 results are not always distinguishable, as shown on the bottom example of Fig. 5.

Fig. 6 compares between GIST1 and Poisson editing, and shows the benefit in optimizing over an overlap region of the two images. In this case, GIST1 overcomes the misalignment between the images, whereas Poisson editing suffers from artifacts.

We note that, while in these examples, there are significant differences between the solution obtained by these methods, in many practical cases, the differences are less noticeable. In the following set of experiments, we demonstrate these methods in various applications including mosaicing, object blending, and reduction of compression artifacts.

A. Stitching Panoramic Views

Stitching results of panoramic views are shown in Figs. 4, 7, and 8.

The input images were captured from different positions and were aligned by a two-dimensional parametric transformation. The aligned images contained local misalignments due to parallax, and photometric inconsistencies due to differences in illumination and in camera settings. Fig. 4 compares gradient methods versus image domain methods. Figs. 7 and 8 shows the results of the stitching algorithms when the input images are misaligned. In all our experiments, GIST1 under ℓ_1 gave the best results, in some cases, comparable with other methods. In Fig. 7, comparable with Feathering, and, in Fig. 8, comparable with "optimal seam." When the input images were misaligned along the seam, GIST1 under ℓ_1 was superior to Poisson editing [1].

B. Stitching Object Parts

Here, we consider combining images of objects of the same class having different appearances. Objects parts from different images are combined to generate the final image. This can be used, for example, by the police in the construction of a suspect's composite portrait from parts of faces in the database. Fig. 10 shows an example for this application, where GIST1 is compared to pyramid blending in the gradient domain. Another example for combination of image parts is shown in Fig. 9.
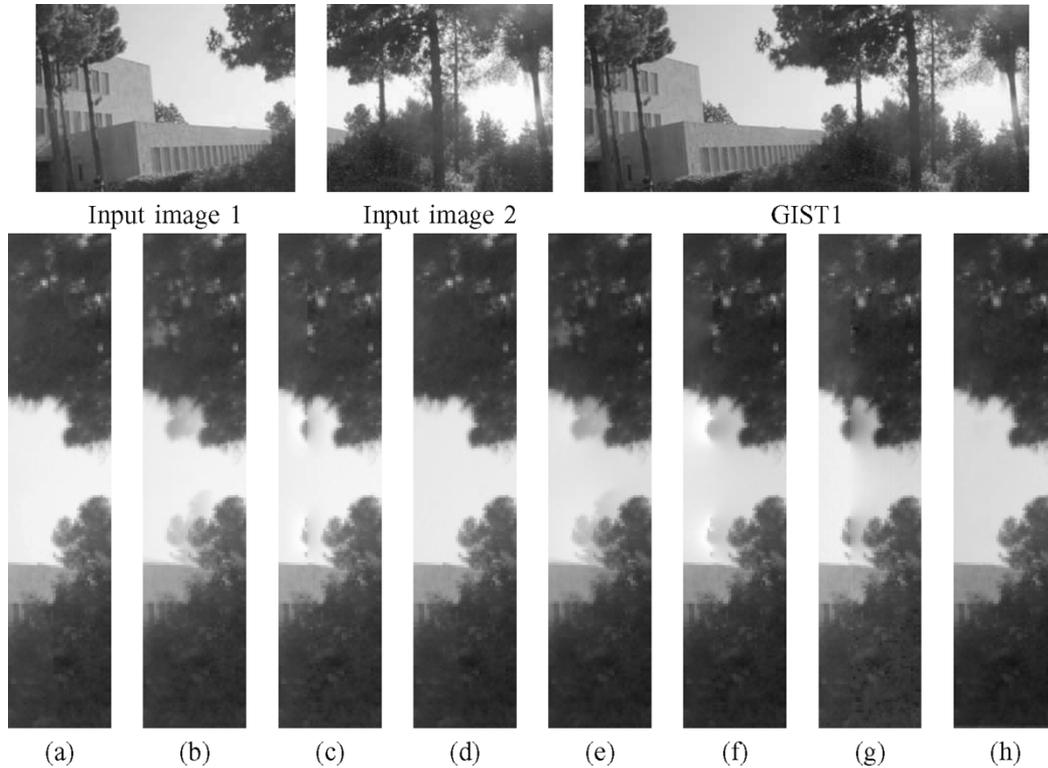
Fig. 8. Comparison between various image stitching methods. On the top are the input image and the result of GIST1 under ℓ_1. The images on the bottom are cropped from the results of various methods. (a) Optimal seam, (b) feathering, (c) pyramid blending, (d) optimal seam on the gradients, (e) feathering on the gradients, (f) pyramid blending on the gradients, (g) Poisson editing [1], and (h) GIST1—ℓ_1. When there are large misalignments, optimal seam and GIST1 produce less artifacts.
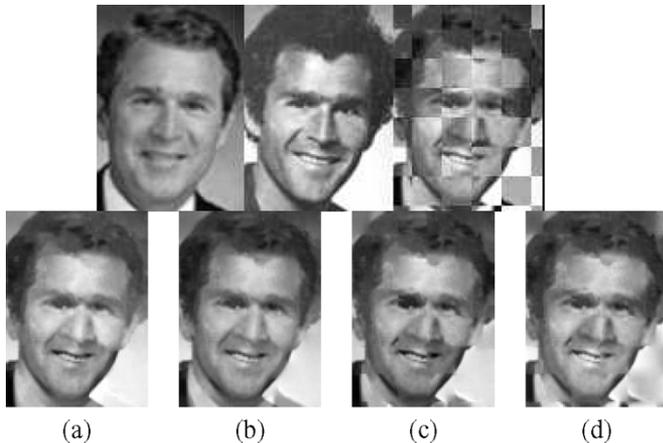

Fig. 9. Combination of images of George W. Bush taken at different ages. On top are the input images and the combination pattern. On the bottom, from left to right, are the results of GIST1 Stitching under ℓ_1 (a) and under ℓ_2 (b), the pyramid blending in the gradient domain (c), and pyramid blending in the image domain (d). Note the duplication of the shoulder in (b), (c), and (d).

Fig. 10. Police application for generating composite portraits. The top row shows the image parts used in the composition, taken from the Yale database. The bottom row shows, from left to right, the results of pasting the original parts, GIST1 under ℓ_1, GIST1 under ℓ_2, and pyramid blending in the gradient domain. Note the discontinuities in the eyebrows.

C. Object Insertion

In this application, objects are to be inserted seamlessly into an image. Our method overcomes photometric inconsistencies as well as small misalignments. Fig. 11 shows an example where an image of woman with closed eyes is fixed by replacing the facial area. The inserted facial part was captured at a different head orientation, which causes misalignments between the inserted and the original face image. Still, the algorithm managed to create a seamless result. In this example, although there are

misalignments, the differences between the results of Poisson editing and Gist were hardly visible (not shown).

D. Removing Compression Artifacts

Lossless block-based compression methods, like JPEG, generate images with gradient artifacts along the block boundaries. A modified version of GIST1 was applied to highly compressed JPEG images in order to reduce block artifacts, by minimizing

$$d_p(\nabla I, I_0, \cup_i, \pi, W) + k_2 \sum_i \sum_{\vec{q} \in \psi_i} |I(\vec{q}) - \mathrm{DC}_i| \quad (4)$$

with respect to I. Here, I_0 is the compressed input image, ψ_i is the ith image block, W is uniform except block boundaries,

Fig. 11. Object insertion. (a) Input image that was captured while the woman was blinking. We copied the top part of the facial region from another image, where the woman's head had a different orientation. (b) Input image with the inserted part overlaid. The red line marks the mask boundaries. (c) Results of GIST1 under ℓ_1. Similar results were obtained with Poisson editing [1].
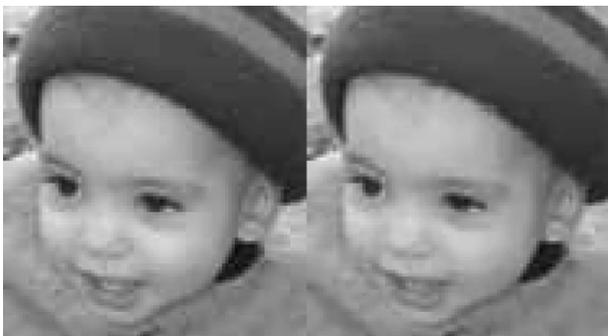
Fig. 12. Reduction of compression artifacts. See text for details.

where it has a low value k_1, and DC_i is the DC component of block i. The parameters k_1, k_2 can be tuned to control the result smoothness. Results are shown in Fig. 12.

V. DISCUSSION

In this paper, we studied different methods for image stitching, and focused on a novel approach that optimizes over image derivatives. We explored the differences between the proposed approach and alternative methods, and the differences in optimizing under different norms. Even though each stitching algorithm works better for some images and worse for others, we found that GIST1 under ℓ_1 always worked well. The use of the ℓ_1 norm was especially valuable in overcoming geometrical misalignments of the input images. The drawback of optimizing under ℓ_1 is computational, as it is two orders of

magnitudes slower than minimizing under ℓ_2. Therefore, since, in many cases, the results of GIST1 under ℓ_1 is comparable with other methods, we recommend to use this method only when faster methods fail.

Image stitching was presented as a search for an optimal solution to an image quality criterion. Encouraged by the results obtained by this approach, we believe that it will be useful to explore alternative criteria for image quality using additional image features and results on statistics of natural images [23], [24].

REFERENCES

[1] P. Perez, M. Gangnet, and A. Blake, "Poisson image editing," in *Proc. SIGGRAPH*, Jul. 2003, pp. 313–318.
[2] W. Freeman, E. Pasztor, and O. Carmichael, "Learning low-level vision," in *Proc. Int. Conf. Computer Vision*, 1999, pp. 1182–1189.
[3] D. Milgram, "Computer methods for creating photomosaics," *IEEE Trans. Comput.*, vol. C-23, no. , pp. 1113–1119, 1975.
[4] A. Efros and W. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. SIGGRAPH*, Aug. 2001, pp. 341–346.
[5] J. Davis, "Mosaics of scenes with moving objects," in *Proc. Conf. Computer Vision and Pattern Recognition*, 1998, pp. 354–360.
[6] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen, "Interactive digital photomontage," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 294–302, 2004.
[7] S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet, "Mosaicing on adaptive manifolds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1144–1154, Oct. 2000.
[8] M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating ghosting and exposure artifacts in image mosaics," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2001, pp. II:509–II:516.
[9] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and O. J. M., "Pyramid method in image processing," *RCA Eng.*, vol. 29, no. 6, pp. 33–41, 1984.
[10] S. Peleg, "Elimination of seams from photomosaics," *Comput. Graph. Image Process.*, vol. 16, no. 1, pp. 90–94, May 1981.
[11] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *Proc. SIGGRAPH*, Jul. 2002, pp. 249–356.
[12] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Trans. Image Process.*, vol. 10, no. 8, pp. 1200–1211, Aug. 2001.
[13] Y. Weiss, "Deriving intrinsic images from image sequences," in *Proc. Int. Conf. Computer Vision*, 2001, pp. II:68–II:75.
[14] M. Tappen, W. Freeman, and E. Adelson, "Recovering intrinsic images from a single image," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2002, vol. 15.
[15] G. Finlayson, S. Hordley, and M. Drew, "Removing shadows from images," in *Proc. Eur. Conf. Computer Vision*, 2002, p. IV:823.
[16] A. Levin, A. Zomet, and Y. Weiss, "Learning to perceive transparency from the statistics of natural scenes," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2002, vol. 15.
[17] R. Frankot and R. Chellappa, "A method for enforcing integrability in shape from shading algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 4, pp. 439–451, Jul. 1988.
[18] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992.
[19] R. Vanderbei, "Loqo." [Online]. Available: http://www.orfe.princeton.edu/loqo/. (2000).
[20] V. Chvátal, *Linear Programming*. New York: Freeman, 1983.
[21] R. Vanderbei, *Linear Programming: Foundations and Extensions*, 2nd ed. Norwell, MA: Kluwer, 2001.
[22] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *Proc. Eur. Conf. Computer Vision*, 2004, pp. 377–389.

[23] E. Simoncelli, "Bayesian denoising of visual images in the wavelet domain," *BIWBM*, vol. 18, pp. 291–308, 1999.

[24] M. Wainwright and E. Simoncelli, "Random cascades of Gaussian scale mixtures for natural images," in *Proc. Int. Conf. Image Processing*, 2000, pp. I:260–I:263.

Assaf Zomet received the B.A., M.Sc., and Ph.D. degrees from The Hebrew University of Jerusalem, Jerusalem, Israel, in 1997, 1999, and 2003, respectively.

He is the Director of Research at HumanEyes Technologies, Jerusalem. He was a Research Scientist with the Computer Science Department, Columbia University, New York. His research interests include mosaicing, super resolution, low-level vision, and novel cameras.

Anat Levin received the B.S. degree in mathematics and computer science and the M.Sc. degree in computer science from The Hebrew University of Jerusalem, Jerusalem, Israel, in 2000 and 2001, respectively, where she is currently pursuing the Ph.D. degree in computer science.

She visited Mitsubishi Electric Research Laboratories and Microsoft Redmond Research Laboratories in the summers of 2002 and 2003, respectively. Her main research interests are in low-level vision, object recognition, and multiple-view geometry.

Shmuel Peleg received the B.Sc. degree in mathematics from The Hebrew University of Jerusalem, Jerusalem, Israel, in 1976, and the M.Sc. and Ph.D. degrees in computer science from the University of Maryland, College Park, in 1978 and 1979, respectively.

He has been a faculty member at The Hebrew University of Jerusalem since 1980, where he was the Head of the School of Computer Science from 1989 to 1991. He has held visiting positions at the University of Maryland, New York University, and the Sarnoff Corporation. He has published over 130 technical papers in computer vision. He holds several patents which provided the technical foundations to three startup companies: VideoBrush (USA), Emaki (Japan), and HumanEyes (Israel). His research has covered pyramid representation, image enhancement, motion analysis, and panoramic mosaicing.

Yair Weiss received the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge.

He was Visiting Scientist at the University of California, Berkeley. He is currently an Associate Professor at the School of Computer Science and Engineering, The Hebrew University of Jerusalem, Jerusalem, Israel. His research interests include human and machine vision, machine learning, and error-correcting codes.