

Seamless Image Stitching in the Gradient Domain

Anat Levin Assaf Zomet Shmuel Peleg Yair Weiss

E-Mail: {alevin,zomet,peleg,yweiss}@cs.huji.ac.il

School of Computer Science and Engineering,
The Hebrew University of Jerusalem,
91904, Jerusalem, Israel

Abstract. The quality of image stitching is measured 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 quality of stitching. 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.

This 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.

1 Introduction

Image stitching is a common practice in the generation of panoramic images and applications such as object insertion, super resolution [10] and texture synthesis [6]. 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.

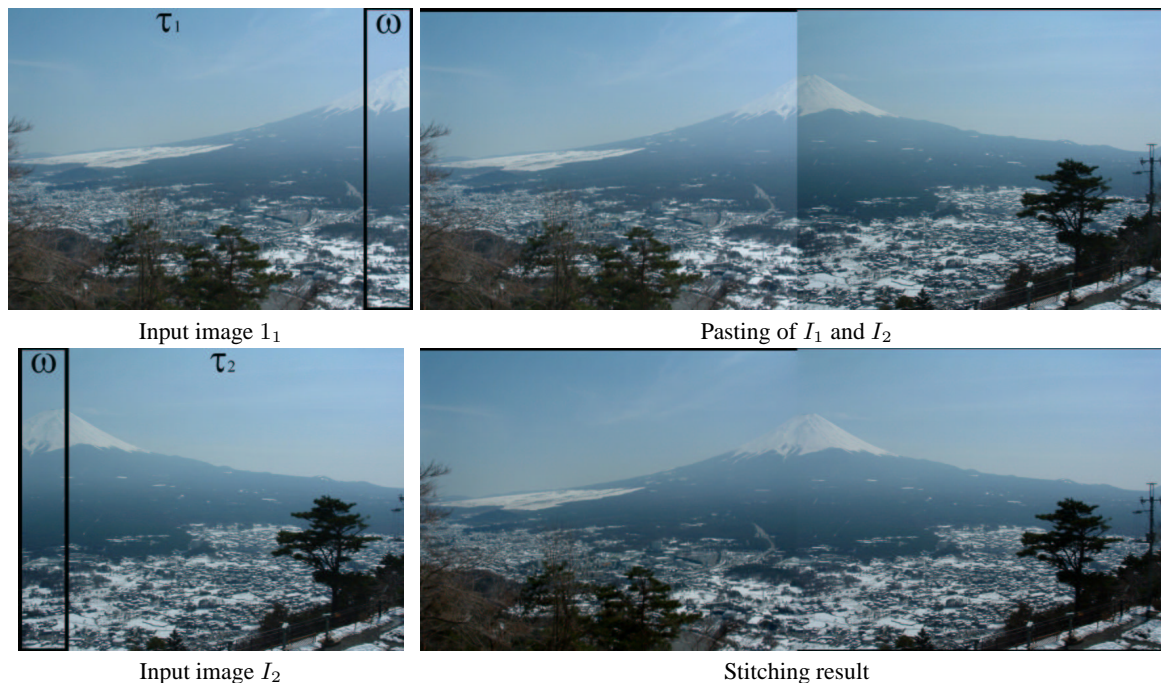


Fig. 1. Image stitching. On the left are the input images. The darkened rectangles mark the overlap region ω . On top right is a simple pasting of the input images. On the bottom right is the result of the GIST1 algorithm.

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.

1.1 Related Work

There are two main approaches to image stitching in the literature, assuming that the images have already been aligned. Optimal seam algorithms [13, 6, 5] 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 is a global intensity difference 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 [15].

The second approach minimizes seam artifacts by smoothing the transition between the images. In Feathering [20] 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 spatially as a function of the distance from the seam. In pyramid blending[1], different frequency bands are combined with different alpha masks. Lower frequencies are mixed over a wide region around the seam, and fine details are mixed in a narrow region around the seam. This produces gradual transition in lower frequencies, while reducing edge duplications in textured regions. A related approach was suggested in [14], 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[20], these methods tend to leave artifacts in the mosaic image such as double edges, as illustrated on the second row of Fig. 2.

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[7], image editing [16], image inpainting [2] and separation of images to layers [23, 19, 8, 11]. The closest work to ours was done by Perez et. al. [16], who suggest to edit images by manipulating their gradients. One of the editing application is object insertion, where an object is cut manually from an image, and inserted to a new background image. The insertion is done by solving the Poisson equation for the gradient field of the inserted object, with boundary conditions defined by the background image. In the discussion section we compare our approach to the work by Perez et. al.

Input image I_1	Input image I_2	Feathering	Pyramid blending	Optimal Seam	GIST

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 . Optimal seam methods produce a seam artifact in case of photometric inconsistencies between the images (first row). Feathering and pyramid blending produce double edges in case of horizontal misalignments (second row). With vertical misalignments (third row), the stitching is less visible when Feathering and GIST are used.

2 GIST: Image Stitching in the Gradient Domain

We describe two approaches to image stitching in the gradient domain. Section 2.1 describes GIST1, where the mosaic image is inferred directly from the derivatives of the input images. Section 2.2 describes GIST2, a two-steps approach to image stitching. Section 2.3 compares the two approaches to each other, and with other stitching algorithms.

2.1 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.

Specifically, let I_1, I_2 be 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, as shown in Fig. 1, 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} :

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

where U is a uniform image, and $d_p(J_1, J_2, \phi, W)$ is the weighted distance between the two fields J_1, J_2 on ϕ :

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

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

The first two terms in Eq. 1 express the dissimilarity of the mosaic image to the input images in the respective regions. 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 non-uniformness in the camera photometric response and due to scene shading variations.

The last two terms in Eq. 1 constrain the gradients in the overlap region, penalizing for derivatives which are inconsistent with any of the input images. In image locations where both I_1 and I_2 have low gradients, these terms penalize for high gradient values in the mosaic image. This property is useful in eliminating false stitching edges.

The choice of norm (parameter p) has implications on both the optimization algorithm and the mosaic image. The minimization of E_p (Eq.1) for $p \geq 1$ is convex, and hence efficient optimization algorithms can be used. Section 3 describes a minimization scheme for E_2 by existing algorithms, and a novel fast minimization scheme for E_1 . The mask image W was either a uniform mask (for E_1) or the Feathering mask (for E_2), which is linear with the signed-distance from the seam. 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.

2.2 GIST2: Stitching Derivative Images

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

1. Compute the derivatives of the input images $\frac{\partial I_1}{\partial x}, \frac{\partial I_1}{\partial y}, \frac{\partial I_2}{\partial x}, \frac{\partial I_2}{\partial y}$.
2. Stitch the derivative images to form a field $F = (F_x, F_y)$. F_x is obtained by stitching $\frac{\partial I_1}{\partial x}$ and $\frac{\partial I_2}{\partial x}$, and F_y is obtained by stitching $\frac{\partial I_1}{\partial y}$ and $\frac{\partial I_2}{\partial y}$.
3. Find the mosaic image 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 stage (2) above, any stitching algorithm may be used. We have experimented with Feathering, pyramid blending [1], and optimal seam. For the optimal seam we used the algorithm in [6], finding the curve $x = f(y)$ that minimizes the sum of absolute differences in the input images. Stage (3), the optimization under ℓ_1, ℓ_2 , is described in Section 3. 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 .

2.3 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. However, based on the following analysis in conjunction with the experiments in Section 4, we recommend using GIST1 under ℓ_1 .

In Theorem 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.

Theorem 2 shows an equivalence between GIST1 under ℓ_2 and Feathering of image derivatives (GIST2) under ℓ_2 ¹.

Both theorems 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.

¹ Note that Feathering of image derivatives is different from Feathering of the images.

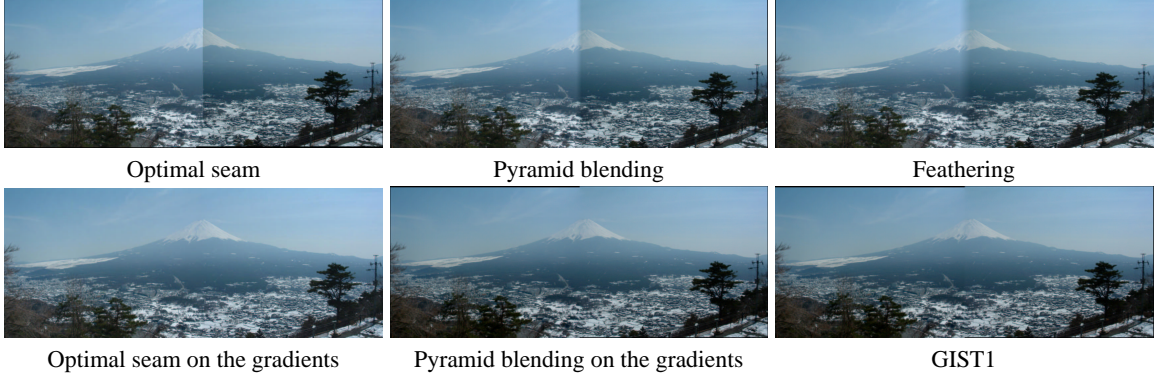


Fig. 3. Stitching in the gradient domain. The input images appear in Fig 1, with the overlap region marked by a black rectangle. With the image domain methods (top panels) the stitching is observable. Gradient-domain methods (bottom panels) overcome global inconsistencies between the images.

Theorem 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 $\mathbf{q} \in \{(f(y), y)\}$, $I_1(\mathbf{q}) = I_2(\mathbf{q})$. Let U be a uniform image. Then the optimal seam solution I , defined below, is a global minimum of $E_p(I; I_1, I_2, U)$ defined in Eq.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 $E_p(I; I_1, I_2, U)$ on the image I is:

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

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

$$E_p(\hat{I}; I_1, I_2, U) = d_p(\nabla \hat{I}, \nabla I_1, \omega, U) + d_p(\nabla \hat{I}, \nabla I_2, \omega, U) + d_p(\nabla \hat{I}, \nabla I_1, \tau_1, U) + d_p(\nabla \hat{I}, \nabla I_2, \tau_2, U) \geq \sum_{\mathbf{q} \in \omega} (\|\nabla \hat{I}(\mathbf{q}) - \nabla I_1(\mathbf{q})\|_p^p + \|\nabla \hat{I}(\mathbf{q}) - \nabla I_2(\mathbf{q})\|_p^p) \geq \sum_{\mathbf{q} \in \omega} \|\nabla I_2(\mathbf{q}) - \nabla I_1(\mathbf{q})\|_p^p = E_p(I; I_1, I_2, U)$$

Result 1 GIST1 under ℓ_1 and the optimal seam method 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 4 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):

Theorem 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 $W(\mathbf{q}) = 1$ for $\mathbf{q} \in \tau_1$, and $W = 0$ for $\mathbf{q} \in \tau_2$). Let I_{Gist} be the minimum of $E_2(I; I_1, I_2, W)$ defined in Eq. 1. Let F be the following field:

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

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 convolution operators, and let $F_{xj} = I_j * f_x, F_{yj} = I_j * f_y$. Then:

$$E_2(I; I_1, I_2, W) = \sum_{\mathbf{q} \in \omega} W(\mathbf{q}) [(I * f_x)(\mathbf{q}) - F_{x1}(\mathbf{q})]^2 + [(I * f_y)(\mathbf{q}) - F_{y1}(\mathbf{q})]^2 + \sum_{\mathbf{q} \in \omega} (1 - W(\mathbf{q})) [(I * f_x)(\mathbf{q}) - F_{x2}(\mathbf{q})]^2 + [(I * f_y)(\mathbf{q}) - F_{y2}(\mathbf{q})]^2$$

Define P_W as the linear operator multiplying each image location \mathbf{q} by a weight $W(\mathbf{q})$, 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 \mathbf{q} :

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

Noting that $P_{1-W} + P_W$ is the identity operator, we get the Poisson equation:

$$\Delta I(\mathbf{q}) = \text{div}(W(\mathbf{q})F_1(\mathbf{q}) + (1 - W(\mathbf{q}))F_2(\mathbf{q}))$$

Where Δ is the Laplacian operator, $F_1 = (F_{x1}, F_{y1}), F_2 = (F_{x2}, F_{y2})$. Hence the minimum of $E_2(I; I_1, I_2)$ is obtained with the image having the closest derivatives to F under 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)

3 Implementation details

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

Eq. 1 defines a set of linear equations in the image intensities, with the derivative filter taps as the equations coefficients. Similarly to [23, 19], we found that good results are obtained using forward-differencing derivative filters $\frac{1}{2} [1 \ -1]$. In the ℓ_1 case, the results were further enhanced by incorporating additional equations using derivative filters in multiple scales. In our experiments we added the filter corresponding to forward-differencing in the 2nd level of a Gaussian pyramid. This filter is obtained by convolving the filter $[1 \ 0 \ -1]$ with a vertical and a horizontal Gaussian filter ($\frac{1}{4} [1 \ 2 \ 1]$). Color images were handled by applying the algorithm to each of the color channels separately.

The minimum to Eq. 1 under ℓ_2 with mask W was shown in theorem 2 to be the image with the closest derivatives to the following field F under ℓ_2 norm :

$$F = \begin{cases} W(\mathbf{q})\nabla I_1(\mathbf{p}) & \mathbf{q} \in \tau_1 \\ W(\mathbf{q})\nabla I_1(x, y) + (1 - W(\mathbf{q}))\nabla I_2(x, y) & \mathbf{q} \in \omega \\ \nabla I_2(x, y) & \mathbf{q} \in \tau_2 \end{cases}$$

The solution can be obtained by various methods, e.g. de-convolution [23], FFT [9] or multi-grid solvers [17]. The results presented in this paper were obtained by FFT.

As for the ℓ_1 optimization, we found using a uniform mask U to be sufficient. Solving the linear equations under ℓ_1 can be done by linear programming as follows [4]:

$$\begin{aligned} \text{Min} : & \sum_i (z_i^+ + z_i^-) \\ \text{Subject to} : & \\ & Ax + (z^+ - z^-) = b \\ & x \geq 0, z^+ \geq 0, z^- \geq 0 \end{aligned}$$

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

The linear program was solved using LOQO, an efficient software package [21]. A typical execution time for a 200×300 image on a Pentium 4 using this package was around 2 minutes. Since no boundary conditions were used, the solution was determined up to a uniform intensity shift. This shift can be determined 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 image.

3.1 Iterative ℓ_1 Optimization

A faster implementation for the ℓ_1 optimization can be achieved by an iterative algorithm in the image domain. We first describe the algorithm when the forward differencing derivatives are used with kernel $\frac{1}{2} [1 \ -1]$, and then generalize to other convolution filters.

Let Dx_j, Dy_j be the forward-differencing derivatives of input image I_j . The iterative ℓ_1 optimization is performed as follows:

- Initialize the solution image I

- Iterate until convergence:
 - for all x, y in the image, update $I(x, y)$ as follows:

$$I(x, y) \leftarrow 2 * \text{median}(\cup_k \{ \begin{matrix} I(x+1, y) - Dx_k(x, y), & I(x-1, y) + Dx_k(x-1, y), \\ I(x, y+1) - Dy_k(x, y), & I(x, y-1) + Dy_k(x, y-1) \end{matrix} \}) \quad (3)$$

For an even number of samples, the median is taken to be the average of the two middle samples. In regions τ_j where a single image I_j is used, the median is taken on the predictions of $I(x, y)$ given its four neighbours and the derivatives of image I_j . For example, when the derivatives of image I_j are 0, the algorithm performs an iterated median filter of the neighbouring pixels. In the overlap region ω of I_1, I_2 , the median is taken over the predictions from both images.

With the forward-differencing derivatives, the algorithm has a simple implementation for parallel architectures. Define a 'checkerboard' partitions of the pixels, with the 'black' locations as $B = \{(x, y) | \text{mod}(x - y, 2) = 0\}$ and the 'White' locations as $W = \{(x, y) | \text{mod}(x - y, 2) = 1\}$. Since in each iteration, the pixels in W are influenced only by pixels in B and vice-versa, the update stage (3) can be applied to all pixels in W in parallel, and similarly to all pixels in B .

The generalization of the algorithm for other kernels is straight-forward. Let $\{f_k(u, v)\}$ be the derivative kernels. Let $D_j^k = I_j * f_k$, where '*' is the convolution operator. The update stage (3) is modified to a weighted median:

$$I(x, y) \leftarrow \text{weightedMedian}_{j,k,(u,v) | f_k(\hat{u}, \hat{v}) \neq 0} \left(\left(\frac{D_j^k(u, v) - \sum_{(m,n) \neq (x,y)} f_k(u-m, v-n) I(m, n)}{f_k(\hat{u}, \hat{v})}, \frac{1}{f_k(\hat{u}, \hat{v})} \right) \right)$$

where $\hat{u} = u - x, \hat{v} = v - y$, and a weighted median of a set of sorted samples $\{s_i\}_{i=1}^N$ and their corresponding weights $\{w_i\}_{i=1}^N$ is defined by:

$$\text{weightedMedian}(s_i, w_i) \equiv s_j | j = \text{argmin}_k \sum_{i \leq k} w_i \geq \frac{1}{2} \sum_{i \leq N} w_i$$

At every iteration, the algorithm performs a coordinate descent and improves the cost function (unless a stationary point is reached). Since the cost function is bounded from below, the algorithm always converges. However, although the cost function is convex, the algorithm does not always converge to the global optimum of the function². To improve the algorithm convergence and speed, we combined it in a multi-resolution scheme. In extensive experiments we have conducted with the multi-resolution extension the algorithm always converged to the global optimum.

4 Experiments

In the experiments we applied GIST to various applications including mosaicing, object blending and reduction of compression artifacts.

First, we compared our method to existing image stitching techniques, which work on the image intensity domain: Feathering [20], Pyramid Blending [1], and 'optimal seam' (Implemented as in [6]). The experiments (Fig. 3) validated the advantage in working in the gradient for overcoming photometric inconsistencies. Second, we examined the various techniques described in Section 2.2 and compared their results with GIST1. See for example Fig. 4, 5.

4.1 Stitching Panoramic Views

The natural application for image stitching is the construction of panoramic pictures from multiple ordinary pictures. There are several sources of inconsistencies between the input images. Geometrical misalignments between input images is caused by lens distortions, by the presence of moving objects, and by motion parallax. Photometric inconsistencies between input images may be caused by a varying gain, by lens vignetting, by illumination changes, specularities etc.

The input images in our experiments were captured from different camera positions, and were aligned by a 2D parametric transformation. Local misalignments were unavoidable due to parallax, and photometric inconsistencies were due to differences in illumination and in camera gain. Results of Panoramic mosaicing are shown in Figs 3,4,5. Fig 3 compares gradient methods vs. image domain methods. Fig. 4,5 demonstrate the performance 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 techniques: In Fig. 4 comparable with Feathering, and in 5 comparable with 'optimal seam'.

² Consider for example an image whose left part is white and the right part is black. When applying the algorithm on the derivatives of this image, the uniform image is a stationary point of the algorithm

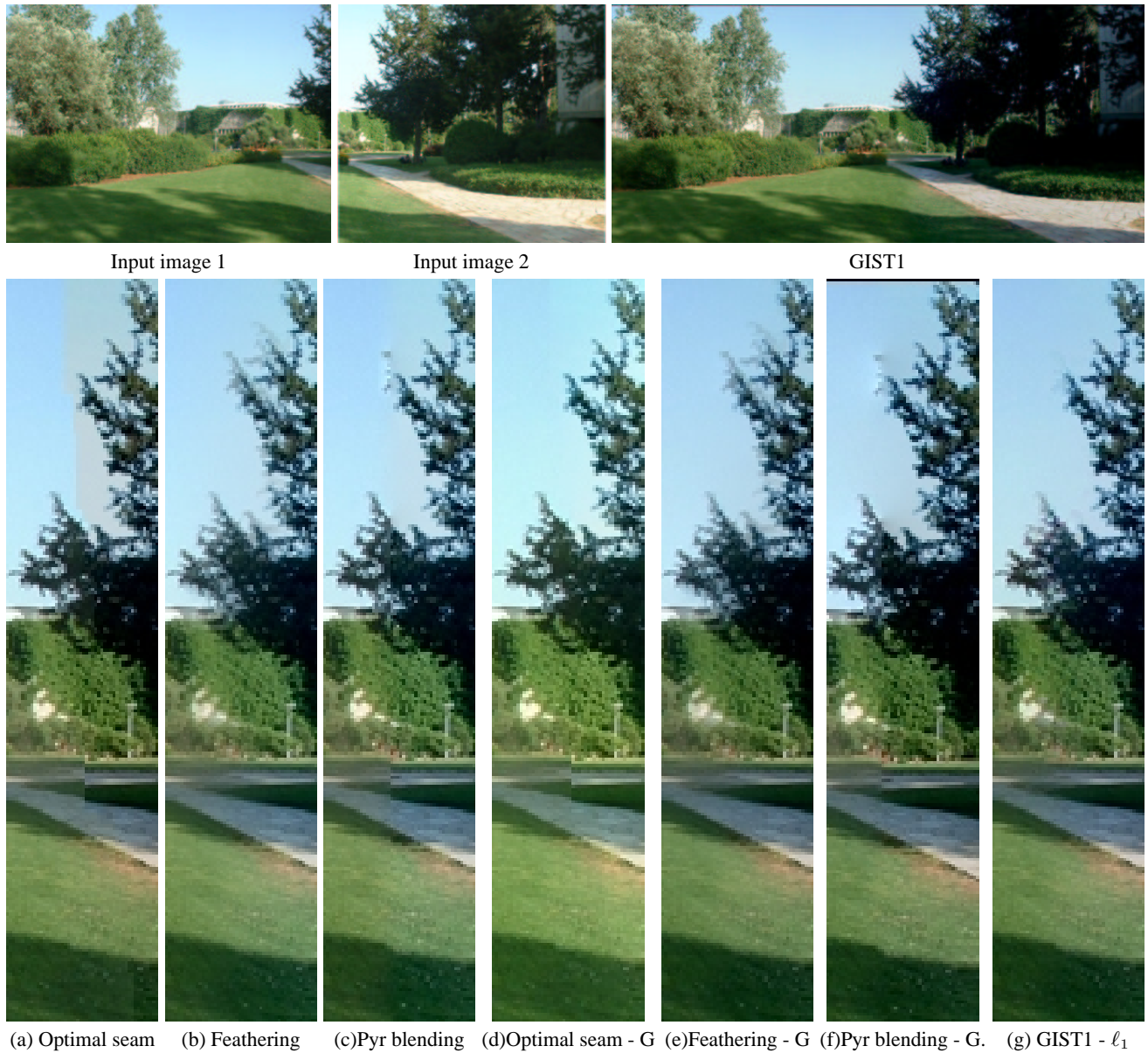


Fig. 4. A comparison between various image stitching methods. On top are the input image and the result of GIST1 under ℓ_1 . The images on bottom are cropped from the results of various methods. (a-c) Image domain methods, (d-f) GIST2, (g) GIST1 - ℓ_1 . The seam is visible in (a),(c) and (d).

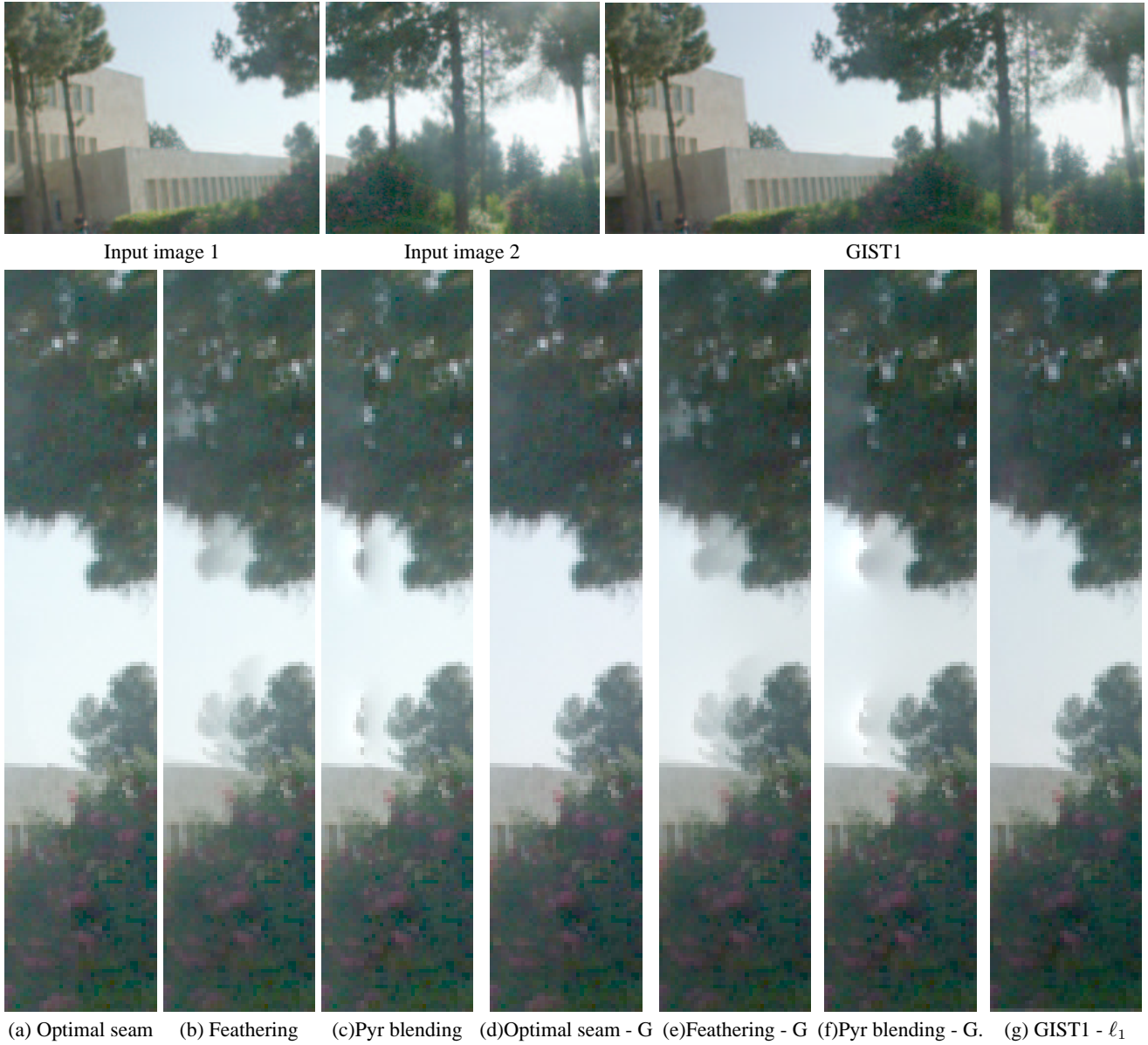


Fig. 5. A comparison between various image stitching methods. On top are the input image and the result of GIST1 under ℓ_1 . The images on bottom are cropped from the results of various methods. (a-c) Image domain methods, (d-f) GIST2, (g) GIST1 - ℓ_1 . When there are large misalignments, optimal seam and GIST1 produce less artifacts.



Fig. 6. A police application for generating composite portraits. The top panel shows the image parts used in the composition. The bottom panel 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.

4.2 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 6 shows an example for this application, where GIST1 is compared to pyramid blending in the gradient domain. Another example for combination of parts is shown in Fig. 8.

4.3 Removing Compression Artifacts

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 the following criterion:

$$E_p(I; I_0) = d_p(\nabla I, I_0, \cup_i, \pi, W) + k_2 \sum_i \sum_{p \in \psi_i} |I(p) - DC_i| \quad (4)$$

Where I_0 is the compressed input image, ψ_i is the i -th image block, W is uniform except block boundaries, where it has a low value k_1 , and DC_i is the DC component of block i . The parameter k_1, k_2 can be tuned to control the result smoothness.

5 Discussion

A novel approach to image stitching was presented, with two main components: First, images are combined in the gradient domain rather than in the intensity domain. This reduces global inconsistencies between the stitched parts due to illumination changes and changes in the camera photometric response. Second, the mosaic image is inferred by optimization over image gradients, thus reducing seam artifacts and edge duplications. Experiments comparing gradient domain stitching algorithms and existing image domain stitching show the benefit of stitching in the gradient domain. Even though each stitching algorithm works better for some images and worse for others, we found that GIST1 under ℓ_1 always worked well and we recommend it as the standard stitching algorithm. The use of the ℓ_1 norm was especially valuable in overcoming geometrical misalignments of the input images.



Fig. 7. Reduction of compression artifacts. See text for details.

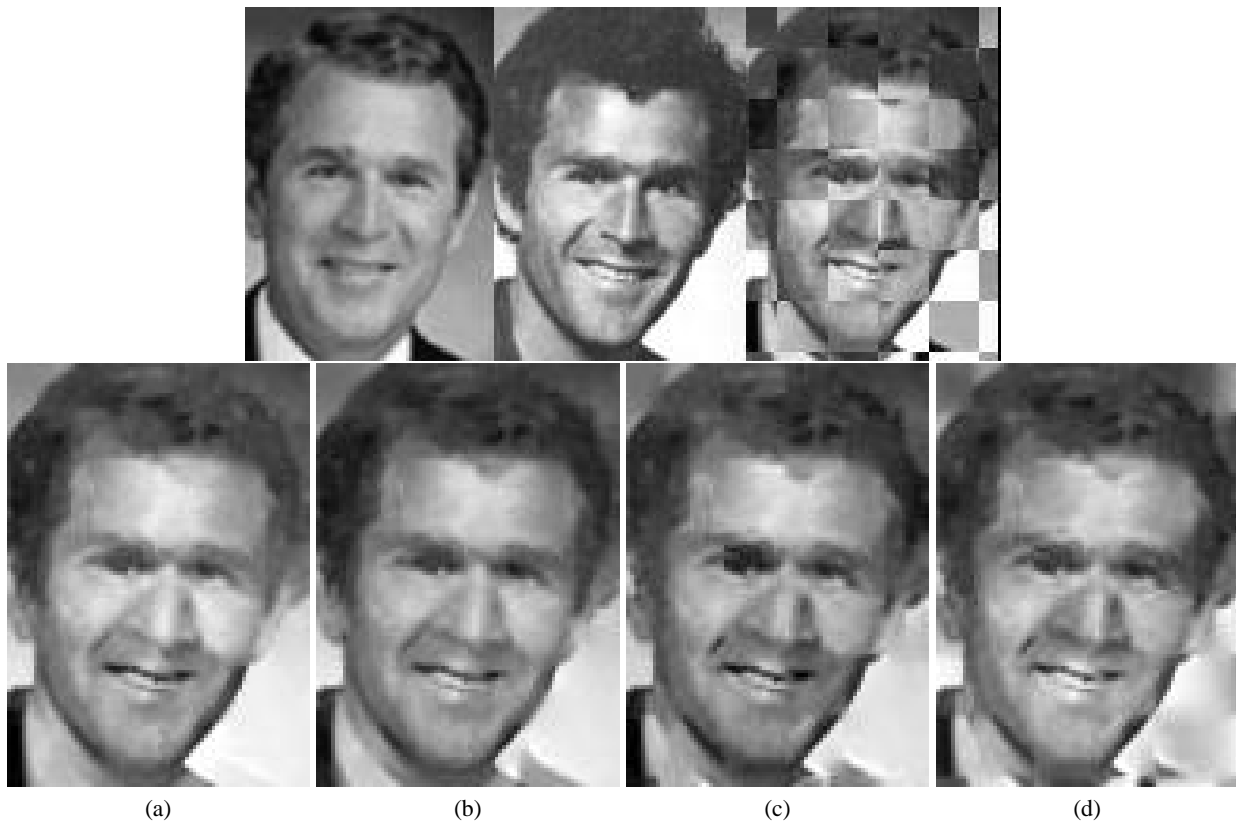


Fig. 8. A combination of images of George W. Bush taken at different ages. On top are the input images and the combination pattern. On the bottom left are, from left to right, the results of GIST1 Stitching under ℓ_1 (a) and under ℓ_2 (b), the results of pyramid blending in the gradient domain (c), and pyramid blending in the image domain (d).

¹² The closest approach to ours was presented recently by Perez et. al. [16] for an application of image editing. There are two main differences with this work: First, in this work we optimize over the gradients of **both** images in the overlap **region**, while Perez et. al. optimized over the gradients of the inserted object and the intensities of the background image. Second, the optimization is done under different norms, while the Poisson equation is defined for the ℓ_2 norm. Both differences have considerable influence on the results, especially in textured regions.

Image stitching was presented as a search for an optimal solution to an image quality criterion. The optimization of this criterion under norms ℓ_1, ℓ_2 is convex, having a single solution. Encouraged by the results obtained by this approach, we believe that it will be interesting to explore alternative criteria for image quality. One direction can use results on statistics of filter responses in natural images [12, 18, 22]. Another direction is to incorporate additional image features in the quality criterion, such as local curvature. Successful results in image inpainting[2, 3] were obtained when image curvature was used in addition to image derivatives.

References

1. E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and Ogden J. M. Pyramid method in image processing. *RCA Engineer*, 29(6):33–41, 1984.
2. 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 Processing*, 10(1), August 2001.
3. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *SIGGRAPH*, July 2000.
4. V. Chvátal. *Linear Programming*. W.H. Freeman and CO., New York, 1983.
5. J.E. Davis. Mosaics of scenes with moving objects. In *Conf. on Computer Vision and Pattern Recognition*, pages 354–360, 1998.
6. A.A. Efros and W.T. Freeman. Image quilting for texture synthesis and transfer. *Proceedings of SIGGRAPH 2001*, pages 341–346, August 2001.
7. R. Fattal, D. Lischinski, and M. Werman. Gradient domain high dynamic range compression. *Proceedings of SIGGRAPH 2001*, pages 249–356, July 2002.
8. G.D. Finlayson, S.D. Hordley, and M.S. Drew. Removing shadows from images. In *European Conf. on Computer Vision*, page IV:823, 2002.
9. R.T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(4):439–451, July 1988.
10. W.T. Freeman, E.C. Pasztor, and O.T. Carmichael. Learning low-level vision. In *Int. Conf. on Computer Vision*, pages 1182–1189, 1999.
11. A. Levin, A. Zomet, and Y. Weiss. Learning to perceive transparency from the statistics of natural scenes. In *Advances in Neural Information Processing Systems*, volume 15. The MIT Press, 2002.
12. S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.
13. D.L. Milgram. Computer methods for creating photomosaics. *IEEE Trans. Computer*, 23:1113–1119, 1975.
14. S. Peleg. Elimination of seams from photomosaics. *Computer Graphics and Image Processing*, 16(1):90–94, May 1981.
15. S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet. Mosaicing on adaptive manifolds. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10):1144–1154, October 2000.
16. P. Perez, M. Gangnet, and A. Blake. Poisson image editing. *Proceedings of SIGGRAPH 2003*, pages 313–318, July 2003.
17. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, Cambridge (UK) and New York, 2nd edition, 1992.
18. E.P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. *BIWBM*, 18:291–308, 1999.
19. M.F. Tappen, W.T. Freeman, and E.H. Adelson. Recovering intrinsic images from a single image. In *Advances in Neural Information Processing Systems*, volume 15. The MIT Press, 2002.
20. M. Uyttendaele, A. Eden, and R. Szeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *Conf. on Computer Vision and Pattern Recognition*, pages II:509–516, 2001.
21. R. Vanderbei. Loqo, <http://www.orfe.princeton.edu/loqo/>, 2000.
22. M.J. Wainwright and E.P. Simoncelli. Random cascades of gaussian scale mixtures for natural images. In *Int. Conf. on Image Processing*, pages I:260–263, 2000.
23. Y. Weiss. Deriving intrinsic images from image sequences. In *Int. Conf. on Computer Vision*, pages II: 68–75, 2001.