

Enforcing Stochastic Inverse Consistency in Non-Rigid Image Registration and Matching *

Sai-Kit Yeung ^{1†} Chi-Keung Tang ¹ Pengcheng Shi ² Josien P.W. Pluim ³
 Max A. Viergever ³ Albert C.S. Chung ¹ Helen C. Shen ¹

¹ The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong

² Rochester Institute of Technology, Rochester, New York, USA

³ Image Sciences Institute, University Medical Center Utrecht, Utrecht, The Netherlands

Abstract

This paper presents a new method to enforce inverse consistency in nonrigid image registration and matching. Conventional approaches assume diffeomorphic transformation, implicitly or explicitly. However, the inherent smoothness constraint discourages discontinuity consideration. We propose a post-processing algorithm that integrates the input forward and backward fields, which are output by existing registration/matching algorithms, to produce more robust results. Given such a pair of input fields, our algorithm alternately refines the fields by tensor belief propagation, and enforces inverse consistency in stochastic sense by generalized total least squares fitting. To show the efficacy of our stochastic inverse consistency approach, we first present results on very noisy fields. We then demonstrate improvement on existing stereo matching where occlusion is naturally handled by localizing violations of inverse consistency. Finally, we propose a novel application on image stitching, where stochastic inverse consistency is employed in structure deformation, in order to seamlessly align overlapping images with severe misalignment in structure and intensity.

Keywords: Image registration and matching.

1. Introduction

In non-rigid image registration of two input images, registering I_1 to I_2 and registering I_2 to I_1 should result in two mappings that are inverses to each other, except at regions of large topological change where no inverses exist.

*This research was supported by the Research Grant Council, Hong Kong (CERG) under grant no. 619208 and K S Lo Foundation.

[†]Part of this work was done while Sai-Kit Yeung was a visiting student at Image Sciences Institute, University Medical Center Utrecht, Utrecht, The Netherlands, under the support of Kopie brief betref Chinese PhD fellowships.

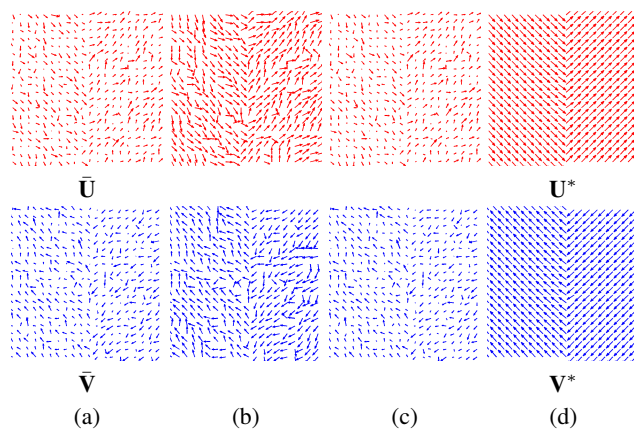


Figure 1. Enforce inverse consistency in the presence of noise and discontinuity. (a) Input noisy fields where each component of the vector is corrupted by Gaussian noise of zero mean and standard deviation 0.7. (b) Refined fields obtained by smoothing the fields respectively using TBP alone. (c) Refined fields obtained by performing GTLS alone on the respective field. (d) Inverse consistent fields computed using our TBP-GTLS, which has little difference from the ground truth.

Integrating the registration results from both directions has been shown to produce more robust results than considering either direction alone ([5, 11, 12, 9, 18, 15, 1, 2] or see Figure 1). All previous works on integrating non-rigid image registrations by symmetrizing forward and backward transform assume the notion of *diffeomorphism*, either implicitly (e.g. as soft constraint in a cost function [5]) or explicitly [11, 1, 15, 2].

1.1. Diffeomorphism

Diffeomorphic transform is, by definition, continuous, one-to-one, onto and differentiable. Many previous works have leveraged diffeomorphism to the advantage of produc-

ing invertible transforms. In [11], the authors enforced diffeomorphism by geodesic interpolating spline [4]. Inverse consistency is achieved by enforcing the geodesic paths of the interpolating spline to be time-symmetric. In [1], a variational energy was defined to explicitly divide the image registration diffeomorphism into two halves such that the source and target images contribute equally to the path. [2] extends the mapping algorithm in [3], and makes it symmetric by introducing two choices of symmetric data terms in the matching cost function, either by symmetrizing the flow of diffeomorphism at each point along the flow, or performing the matching only in the midpoint of the large deformation flows, so that both directions contribute equally. Using the diffeomorphic assumption, the authors in [15] symmetrize the common \mathcal{L}_2 and information theoretic objective functions in nonrigid registration. In [12], the authors assume a B-spline model and add an additional constraint on the positivity of the Jacobian to preserve topology.

The mathematical differentiability of diffeomorphism dictates that the transformation function and its inverse exist and are smooth. While this is often true for medical imagery (where some previous works are related), Figure 1 shows a typical example where the forward and backward field are noisy, and each field contains a discontinuity which is not apparent in the input. If we apply any state-of-the-art algorithms that assume diffeomorphism, the noise as well as the inherent discontinuity will very likely be smoothed out in order to satisfy the implicit smoothness criterion, especially when no landmarks (interpolation constraints) are given in this case. The resulting mappings will be far from the desired ground-truth.

There is one further drawback of diffeomorphism in integrating two discrete input mappings. While symmetry is a necessary condition for points that have a corresponding inverse, in non-rigid image registration, the corresponding points of I_1 can be absent from I_2 , and vice versa. In this case, it is desirable to have these points labeled, instead of fitting an invertible function.

1.2. Stochastic Inverse Consistency

In this paper, we propose a *stochastic* approach to integrate two discrete input vector fields, respectively obtained by registering/matching I_1 to I_2 and I_2 to I_1 . The resultant pair of fields are inverse consistent *stochastically*, except at regions where no reliable inverses are found and these regions are labeled. We assume no diffeomorphism or other model. Our algorithm leverages existing registration/matching algorithms to produce more robust results, by adopting a post-processing approach which integrates the information from both directions. This basic premise we use is called *stochastic inverse consistency*:

Stochastic Inverse Consistency. Let $\bar{\mathbf{V}}$, $\bar{\mathbf{U}}$ be the input forward and backward fields, $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{U}}$ be the correspond-

ing inverses. Let $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{v}}$ be one vector from the corresponding fields, such that the vectors $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$ are for the forward direction, and $\tilde{\mathbf{v}}$ and $\tilde{\mathbf{u}}$ are for the backward direction. By considering the x -dimension (the other dimensions are similar), $\bar{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ are *stochastically inverse consistent* if

$$(\bar{u}_x)(\tilde{v}_x) = u_x v_x + r_x \quad \text{or} \quad (1)$$

$$(u_x + e_{\bar{u}_x})(v_x + e_{\tilde{v}_x}) = u_x v_x + r_x \quad (2)$$

where for the two given error-perturbed vectors, we consider their respective x -components \bar{u}_x , and \tilde{v}_x . Their product is going to be deviated from the product of the ground truth u_x and v_x (with unknown error terms $e_{\bar{u}_x}$, $e_{\tilde{v}_x}$, and r_x). While the above equations are for the forward direction, replacing \bar{u}_x with \tilde{u}_x and \tilde{v}_x with \bar{v}_x results in the equations for the backward direction.

We shall demonstrate that by adopting stochastic inverse consistency, our algorithm is robust to noises, does not smooth out discontinuities, symmetrizes the fields where inverses exist, and labels in the fields where no reliable inverses can be found. In the rest of the paper, we will describe our algorithm in section 2, analyze our algorithm in section 3, and demonstrate in section 4 the enforcement of stochastic inverse consistency can improve stereo matching result. We show a novel application in image stitching where inverse consistency is employed in structure and luminance deformation to produce unbiased seamless stitching results.

2. The stochastic inverse consistency algorithm

Given a pair of vector fields computed using existing registration/matching methods, we adopt a Markov Random Field (MRF) approach to model the neighborhood relationship in the vector fields. Under the MRF model, the input pair of vector fields are iteratively refined by tensor belief propagation (TBP) algorithm [17], which is effective in reducing noise, preserving discontinuities, and recovering missing vectors. The enforcement of the inverse consistency is then achieved by fitting a pair of vector fields using the Generalized Total Least Square (GTLS) [16].

By considering the stochastic properties of both the forward and backward vector fields during GTLS fitting, proper weightings are assigned, and “smart averaging” is performed during the optimization process. The process iterates until convergence when stochastic inverse consistency is established, by alternately refining the evolving vector mappings by TBP and stochastic fitting by GTLS.

We denote I_1 to be the source image and I_2 to be the target image in the *forward* direction, i.e., registering I_1 to I_2 (I_2 being fixed). While in the *backward* direction, I_1 is the target image and I_2 is the source image, i.e., registering I_2 to I_1 (I_1 being fixed). We denote \mathbf{U} and \mathbf{V} as the *ground truth* forward and backward vector fields respectively, so in

practice \mathbf{U} and \mathbf{V} are not known. $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$ are the forward and backward vector fields given by a (nonrigid) image registration/matching algorithms. So $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$ are the input to our method. We first summarize our algorithm as follows:

1. Tensor belief propagation (TBP) is applied to refine both vector fields $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$.
2. Compute inverse transformations, $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$. For every vector in the forward field $\bar{\mathbf{U}}$ (resp. backward field $\bar{\mathbf{V}}$), the corresponding inverse vector in the opposite field $\tilde{\mathbf{V}}$ (resp. $\tilde{\mathbf{U}}$) is searched within a local neighborhood in $\bar{\mathbf{V}}$ (resp. $\bar{\mathbf{U}}$).
3. Stochastic fitting using generalized total least square (GTLS) is applied on the forward (resp. backward) and its computed inverse field.
4. Steps 1 to 3 are performed iteratively until all corresponding vector pairs in the forward and its estimated backward (resp. backward and its estimated forward) fields are inverse consistent, and salient inverse inconsistency, if any, is detected. The final output is \mathbf{U}^* (and its optimized inverse field \mathbf{V}^*).

2.1. Field refinement by tensor belief propagation (TBP)

We adopt the MRF approach and propose to refine the input fields $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$ iteratively by tensor belief propagation [17], where missing data is inferred, noisy data is refined, and discontinuities are preserved.

Given the MRF network induced by the image grid where two neighboring nodes are denoted by s and t , the MRF energy function is:

$$E(\bar{\mathbf{U}}) = \sum_s \|\bar{\mathbf{u}}(s) - \bar{\mathbf{u}}_o(s)\| + \lambda \sum_{t \in \mathbf{N}(s)} \log \left(1 + \frac{\|\bar{\mathbf{u}}(s) - \bar{\mathbf{u}}(t)\|}{2\sigma^2} \right) \quad (3)$$

where the relevant terms are defined in the appendix. The tensor belief propagation algorithm, also in appendix, is used to solve Eq. (3) to refine vectors in $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$.

2.2. Computing inverse fields $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$

Given the forward and backward vector fields $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$ refined by TBP, we want to find a corresponding inverse vector fields, that is, to establish the corresponding inverse mappings $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{U}}$ respectively. Note that in case a corresponding inverse vector cannot be reliably computed, a null vector will be returned. Note, after this stage, $\bar{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ ($\bar{\mathbf{V}}$ and $\tilde{\mathbf{U}}$) are not necessarily inverse consistent.

Given a vector $\bar{\mathbf{u}}(s)$ at site s in $\bar{\mathbf{U}}$, a vector $\tilde{\mathbf{v}}(s)$ which represents the inverse of $\bar{\mathbf{u}}(s)$ is to be found from $\bar{\mathbf{V}}$. The

simplest solution is to look at the rounded grid position $p = s + \bar{\mathbf{u}}(s)$ in $\bar{\mathbf{V}}$, and check whether this vector $\bar{\mathbf{v}}(p)$ points toward s in $\bar{\mathbf{U}}$. If it is the case, then, $\tilde{\mathbf{v}}(s) = \bar{\mathbf{v}}(p)$.

However, in many situations such vector cannot be found, because noise and the discrete nature of the data will render this simple point-to-point operation fail to produce reasonable candidate matches. Instead, we adopt neighborhood searching and averaging, such that the chance of finding a reliable inverse is higher while some noise effect can be effectively smoothed out at the same time. The approach is summarized and described in Fig. 7 in appendix.

2.3. GTLS formulation for stochastic inverse consistency

Recall *stochastic inverse consistency*:

$$\begin{aligned} (\bar{u}_x)(\tilde{v}_x) &= u_x v_x + r_x \\ (\tilde{v}_x)(\bar{u}_x) &= u_x v_x + r_x \end{aligned}$$

Note that both the observations (left-hand side) and the model (right-hand side) are perturbed by noise, and that the stochastic property is not the same for \bar{u}_x and \tilde{v}_x . To solve the problem while simultaneously considering both errors, the Generalized Total Least Square (GTLS) ([16], also in appendix) is adopted. Assuming in the current iteration where tensor belief propagation has been executed and inverse fields has been computed, that is, $\bar{\mathbf{V}}$, $\bar{\mathbf{U}}$, $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{U}}$ are available for the current iteration, the x -component is solved at every grid position $s | \tilde{\mathbf{v}}(s)^{(i)} \neq \emptyset$ in the field:

$$\begin{bmatrix} \bar{u}_x(s)^{(i)} \\ -\tilde{v}_x(s)^{(i)} \end{bmatrix} X \approx \begin{bmatrix} -\bar{u}_x(s)^{(i)} \tilde{v}_x(s)^{(i)} \\ -\tilde{v}_x(s)^{(i)} \bar{u}_x(s)^{(i)} \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} \tilde{v}_x(s)^{(i)} \\ -\bar{u}_x(s)^{(i)} \end{bmatrix} Y \approx \begin{bmatrix} -\tilde{v}_x(s)^{(i)} \bar{u}_x(s)^{(i)} \\ -\bar{u}_x(s)^{(i)} \tilde{v}_x(s)^{(i)} \end{bmatrix} \quad (5)$$

where i is the iteration number. X and Y become $\bar{u}_x(s)^{(i+1)}$ and $\tilde{v}_x(s)^{(i+1)}$ respectively. $\tilde{v}_x(s)^{(i+1)}$ and $\bar{u}_x(s)^{(i+1)}$ are then obtained by the corresponding inverse field establishment in the next iteration. Intuitively, Eq. (4) performs fitting such that X will be somewhere in-between $\bar{u}_x(s)^{(i)}$ and $-\tilde{v}_x(s)^{(i)}$. The position depends on their error properties $E_{\bar{u}_x(s)}$ and $E_{\tilde{v}_x(s)}$ which are estimated based on their local variances:

$$\eta = |\bar{u}_x(s)^{(i)} - (-\tilde{v}_x(s)^{(i)})| \quad (6)$$

$$\gamma = \frac{\text{var}(\bar{u}_x(s)^{(i)})}{\text{var}(\bar{u}_x(s)^{(i)}) + \text{var}(\tilde{v}_x(s)^{(i)})} \quad (7)$$

$$E_{\bar{u}_x(s)} = \gamma \cdot \eta, \quad E_{\tilde{v}_x(s)} = (1 - \gamma) \cdot \eta \quad (8)$$

where the error properties on the term $-\bar{u}_x(s)\tilde{v}_x(s)$ are defined by $R = E_{\bar{u}_x(s)}E_{\tilde{v}_x(s)}$ (fitting is not performed if the denominator is zero in Eq. (7)). The error equilibration matrices for solving X in Eq. (4) are obtained from the Cholesky decomposition of the error covariance matrices C and D ,

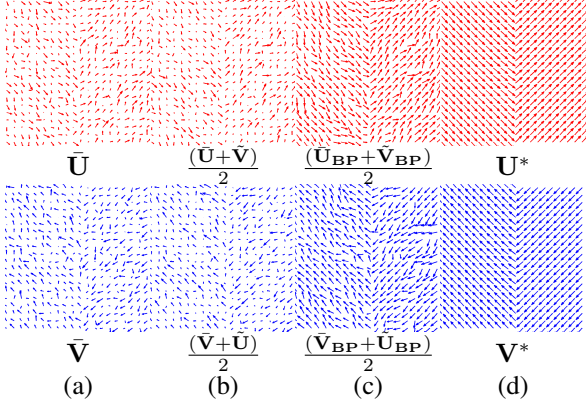


Figure 2. Synthetic data 1: field with discontinuity. (a) The noise-perturbed field with $\sigma_f = \sigma_b = 0.7$. (b) Averaging with computed inverse field. (c) Averaging with results refined by TBP. $\tilde{\mathbf{V}}_{\text{BP}}$ is computed from $\tilde{\mathbf{U}}_{\text{BP}}$ followed by of TBP refinement. $\tilde{\mathbf{U}}_{\text{BP}}$ is similarly computed. (d) Our final result shows little difference from the ground truth.

where $C = \Delta^T \Delta$, $D = \Delta \Delta^T$, $\Delta = \begin{bmatrix} E_{\tilde{u}_x(s)} & R \\ E_{\tilde{u}_y(s)} & R \end{bmatrix}$. Similarly Y is solved. The whole iterative process will be continued until:

$$\sum_{\{s|\tilde{\mathbf{v}}(s)^{(i)} \neq \emptyset\}} \|\tilde{\mathbf{u}}(s)^{(i)} - \tilde{\mathbf{v}}(s)^{(i)}\| + \sum_{\{s|\tilde{\mathbf{u}}(s)^{(i)} \neq \emptyset\}} \|\tilde{\mathbf{v}}(s)^{(i)} - \tilde{\mathbf{u}}(s)^{(i)}\| < \epsilon_{all} \quad (9)$$

where ϵ_{all} is a small constant. Notice that the fitting process and convergence evaluation are omitted at the positions of ‘holes’ where reliable inverses are absent. If the inverse does not exist, the vector should remain unchanged. So the vectors at those positions will only be modified during the TBP process in the next iteration. After the whole process has converged, regions where ‘holes’ still survive in step 2 will be labeled as salient inconsistency region. Experiments show that the number of holes is monotonically decreasing during the process.

3. Analysis of stochastic inverse consistency

First, we use synthetic input vector fields corrupted with various amount of noise. We compare our result with the ground truth (Figure 2 and Table 1). Then, we show how we detect the salient inverse inconsistency (Figure 3).

3.1. Noisy input fields with discontinuity

We first tested our proposed framework on synthetic noisy vector fields with a salient discontinuity to demonstrate the robustness of our framework against noises and the capability of recovering reasonable results close to the ground truths (Figure 2). Different amounts of noise are

σ_f	σ_b		Vector fields with discontinuity					
	0.5	0.6	0.7	0.8				
0.5	(a) $\ \tilde{\mathbf{U}} - \mathbf{U}\ $	(b) $\ \frac{(\tilde{\mathbf{U}}-\mathbf{V})}{2} - \mathbf{U}\ $	(c) $\ \frac{(\tilde{\mathbf{U}}_{\text{BP}}+\tilde{\mathbf{U}}_{\text{BP}})}{2} - \mathbf{U}\ $	(d) $\ \mathbf{U}^* - \tilde{\mathbf{U}}\ $	(e) $\ \tilde{\mathbf{V}} - \mathbf{V}\ $	(f) $\ \frac{(\tilde{\mathbf{V}}-\tilde{\mathbf{U}})}{2} - \mathbf{V}\ $	(g) $\ \frac{(\tilde{\mathbf{U}}_{\text{BP}}+M\tilde{\mathbf{V}}_{\text{BP}})}{2} - \mathbf{V}\ $	(h) $\ \mathbf{V}^* - \tilde{\mathbf{V}}\ $
	257.0	258.2	248.8	306.9	261.8	362.5	257.9	417.3
	254.9	272.4	272.8	277.3	279.3	322.8	297.5	341.0
	96.0	96.8	106.7	116.4	125.5	141.5	101.1	165.1
25.3	26.1	41.7	40.8	37.0	36.0	23.3	23.1	
0.6	298.9	271.6	322.9	311.1	300.9	361.0	330.6	406.5
	283.4	266.8	303.9	289.5	294.3	292.5	324.3	332.6
	119.5	108.2	131.4	121.5	135.0	155.3	172.8	163.3
	96.7	95.9	42.1	42.1	44.7	43.5	42.0	40.7
0.7	360.6	266.9	359.8	311.5	354.7	362.9	372.9	401.3
	309.6	297.4	317.8	299.0	312.8	327.0	323.5	349.2
	132.0	110.9	157.0	132.2	146.9	161.5	160.4	176.3
	24.2	23.5	41.3	38.6	48.7	50.0	83.7	81.9
0.8	408.0	261.8	406.0	313.2	431.4	361.6	423.7	417.2
	335.2	291.1	341.6	329.6	344.9	323.2	350.1	357.0
	165.1	124.9	159.4	137.4	199.9	181.2	200.1	210.9
	47.1	48.5	42.8	43.5	60.1	60.2	87.1	90.5

Table 1. The entries in the bottom table are explained by the table in the top. Ground truth vector fields are \mathbf{U} and \mathbf{V} with $\|\mathbf{U}\| = \|\mathbf{V}\| = 512$. In the top table: σ_f - noise level on perturbed input $\tilde{\mathbf{U}}$, σ_b - noise level on $\tilde{\mathbf{V}}$. Note the error entry in (d) is smallest among (a)–(d), and also (h) among (e)–(h). This table is color-coded.

added to each dimension independently. While simple averaging are unsatisfactory in all the noise levels, simple averaging with TBP incorporated produces result far worse than ours. Table 1 shows the quantitative comparison. We compute the sum of absolute differences between the noisy fields with the ground truth. Let the signal strength of the ground truth field be $\|\mathbf{U}\|$. Excluding the boundary, $\|\mathbf{U}\| = (400 - 38)\sqrt{2} = 512$ as each color is a unit vector. The terms $\|\mathbf{U} - \tilde{\mathbf{U}}\|$ and $\|\mathbf{V} - \tilde{\mathbf{V}}\|$ respectively show the exact errors of the input vector fields. $\|\mathbf{U} - \mathbf{U}^*\|$ and $\|\mathbf{V} - \mathbf{V}^*\|$ measure the error of our recovered vector fields compared with the ground truth. From the table, we can see that even under large amount of noise (i.e., SNR = 1.64 dB for $\|\mathbf{U} - \tilde{\mathbf{U}}\|$ when σ_f and $\sigma_b = 0.7$), our method still recovers the fields with high SNR (SNR = 15.4 dB in the same case).

3.2. Noisy input fields where inverses may not exist

We tested our framework on input field where parts of the vector fields have no inverse. Figure 3 shows the modified forward fields \mathbf{U} and $\tilde{\mathbf{U}}$, where the modified region has no corresponding inverse in the backward field. The results we obtained are similar to the previous example, except that the total number of holes is no longer zero upon convergence. The computed inverse from the final result \mathbf{U}^* and \mathbf{V}^* shows a region of null vectors which actually corresponds to the region with no inverse. This feature is useful for occlusion detection in stereo matching, demonstrated in the following section.

4. Applications

We apply our stochastic inverse consistency to stereo matching in the presence of occlusions, and show that more

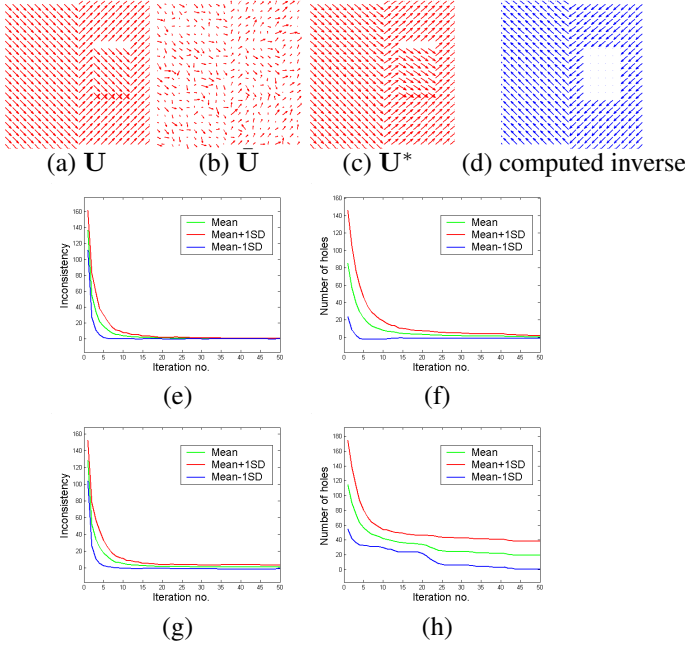


Figure 3. Synthetic data 2: field with discontinuity and non-invertible region. (a) Ground truth. (b) Input forward vector field. The corresponding backward vector fields \mathbf{V} and $\tilde{\mathbf{V}}$ remain unchanged as in Synthetic Data 1. (c) Our result \mathbf{U}^* . (d) The computed inverse from the pair of converged deformation fields. Notice the region where corresponding inverse cannot be found. (e)-(h) compare the convergence behavior between synthetic data 1 and 2 of our algorithm. (e) and (f) are the plot of $\epsilon_{all}/2$ and (total no. of holes)/2 for synthetic data 1, ϵ_{all} is defined in Eq. (9). Correspondingly (g) and (h) are for synthetic data 2. While for both cases they converge quickly, in (h), the total number of holes drops to a constant, rather than zero, as depicted in (f). It shows the fact that upon convergence, there exists region where inverses cannot be reliably computed. The region is automatically labeled by the null vectors as shown in (d).

accurate disparity can be obtained by detecting occlusion as violations in inverse consistency. Finally, we apply our method to general image registration, and show results in image stitching and unconventional image compositing, where feature matching is incorporated to deal with large deformation, in order to align image intensity and preserve underlying image structures without bias.

4.1. Symmetric stereo matching

We demonstrate in this section the enforcement of stochastic inverse consistency improves the results of stereo matching, by symmetrizing the disparity mapping from one image to another. Occlusion is one of the major challenges in stereo matching. Many MRF approaches have already modeled pixel matching, visibility, occlusion constraints in their corresponding energy functional by considering both matching directions [14, 6]. Very good results have been

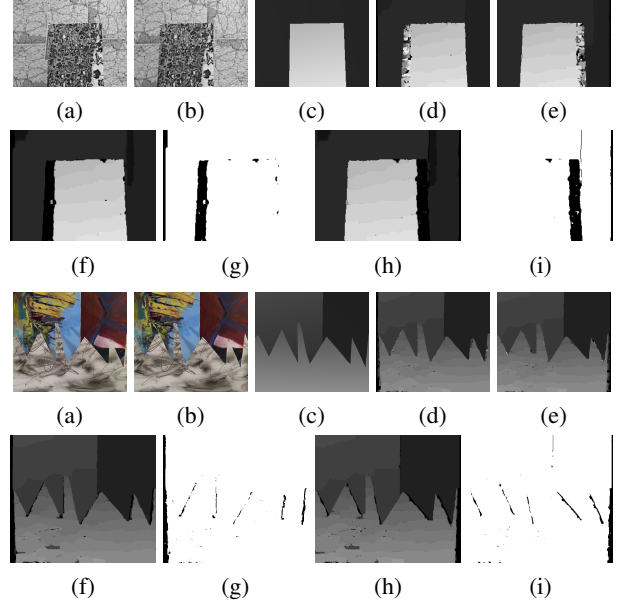


Figure 4. Symmetric stereo results for *Map* (top) and *Sawtooth* (bottom). (a) I_1 . (b) I_2 . (c) Ground truth disparity map, taking I_1 as reference, denoted as \mathbf{U} (forward direction). (d) Forward disparity map $\tilde{\mathbf{U}}$. (e) Backward disparity map $\tilde{\mathbf{V}}$. (f) Refined forward disparity map \mathbf{U}^* by taking $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ as the input. (g) The forward occlusion map, i.e., taking I_1 as reference, detected as violation of inverse consistency using our method. (h) Refined backward disparity map \mathbf{V}^* . (i) The backward occlusion map.

obtained, but the disparity maps obtained by swapping the images may still not be consistent, because each term contributes only a part to the overall energy. Another approach proposed the symmetric Potts model [19], which is a symmetric cost function, and did not explicitly model occlusion.

Enforcing stochastic inverse consistency in stereo matching can be regarded as a data-driven approach to refine a pair of disparity maps, which are obtained by any stereo algorithm by swapping the input images. During the GTLS fitting process, occlusion and depth discontinuities are detected as salient violation to inverse consistency. We demonstrate the idea by applying our method using the stereo framework created by Scharstein and Szeliski [13], which can be found at <http://www.middlebury.edu/stereo>.

Figure 4 shows the improved disparity maps of the *Map* and *Sawtooth* examples. The input disparity maps are computed by the Graph Cuts algorithm [8] provided in Scharstein and Szeliski's package. Figure 4 (d) and (e) show their results, which in turn become our input. From Figure (d) and (e) we can observe that the disparities that significantly deviate from the ground truth (e.g. the noisy disparities in *Map*) are due to occlusion, where matching correspondences are lacking. Our framework refines the disparity maps from both directions with discontinuity consideration. Violation of inverse consistency will be regarded as

occlusion. Notice that our goal is not on introducing a new stereo matching algorithm; rather our contribution here is to simultaneously refine the results obtained from any reasonable stereo algorithm and detect scene occlusion. Note in *Sawtooth* where potential occlusion regions in the disparities are detected.

4.2. Image stitching with large discrepancy in structure and intensity

Global registration is often preferred to pairwise registration in constructing image mosaics in order to minimize accumulation error in registering more than two overlapping images. Here, we address a new problem of stitching multiple images with large discrepancy in structure and intensity, where the images may not capture the same scene or object. Besides these new challenges, similar to conventional mosaic construction, to produce a seamless stitching result, a global method is preferred to avoid accumulation error or biased results. In the following, we shall show how our post-processing approach to enforcing inverse consistency can naturally achieve the following goal: Let I_1, I_2, \dots, I_n be the image sequence to be stitched, where successive images I_i and I_{i+1} are pairwise-overlapping. The same unbiased result is desired when we reverse the stitching sequence, that is, I_n, I_{n-1}, \dots, I_1 .

Existing image stitching algorithms that eliminate across the input images luminance discrepancy [10] and structure mismatch [7] deform input images to a reference image to solve the misalignment problems, therefore generating results dependent on the input order. Our approach to computing an unbiased result is to deform all input images, so that all structures and luminance can be aligned across the stitched result. This is a property often desirable in image stitching.

In order to generate an unbiased stitching result, we deform both input fields under our new inverse consistency framework. The detailed steps are as follows:

1. Similar to stitching algorithms [10, 7], given two images that are roughly aligned, we compute the optimal partition in the overlapping region. Then, detect and match 1D features along the stitching boundary to form a sparse set of deformation vectors, similar in [7].
2. Propagate the deformation vectors to the entire overlapping region. This is different from [7] where the deformation propagation will not be applied across the stitching boundary. The propagation is performed in both the forward and backward directions.
3. The forward and backward vector fields obtained in step 2 become the input of our TBP-GTLS framework. The iterative refinement is performed until a pair of inverse-consistent vector fields is obtained.

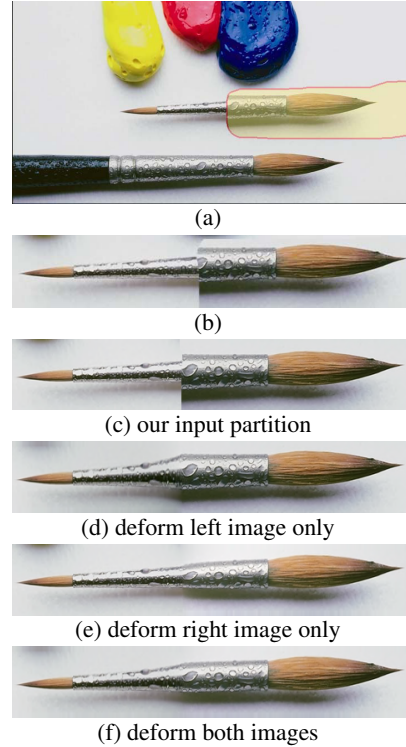


Figure 5. Brush example. Example in [7] for comparison of conventional stitching methods. (a) is the input image. (b) Optimal seam result as shown in [7]. Since we cannot replicate the yellow region directly, for fair comparison we use a new input partition as shown in (c). (d) and (e) are results by [7] where only the left side or the right side is deformed, respectively, showing that the shape transition is not smooth across the stitching boundary, especially in (d). (f) is our result where both sides are deformed to the mean shape. Our result is better since both images are deformed, the shape and luminance transition are smoother. [7] also performed exhaustive comparison using the same example.

Since we cannot replicate the overlapping region as exactly as in [7], for fair comparison, we form a new partition of the two brushes as shown in Figure 5(c). Stitching results for the forward and backward directions are shown in Figure 5(d) and Figure 5(e). In Figure 5(d), it is obvious that a smooth transition of the shape cannot be obtained. Our result is shown in Figure 5(f), where a smooth deformation for both overlapping images is obtained.

Next, we show a complex stitching example of a photomontage created by stitching five images of Frank Lloyd Wright’s architectural masterpieces (Figure 6(a)). Three images are real photos and two are artist’s renderings. The discrepancy in structure and colors are evident as shown in the globally aligned and blended result in Figure 6(b). Note in particular the curved ramps (Guggenheim Museum) which necessitates nonrigid structure deformation here. Figure 6(b) shows the stitching result generated by

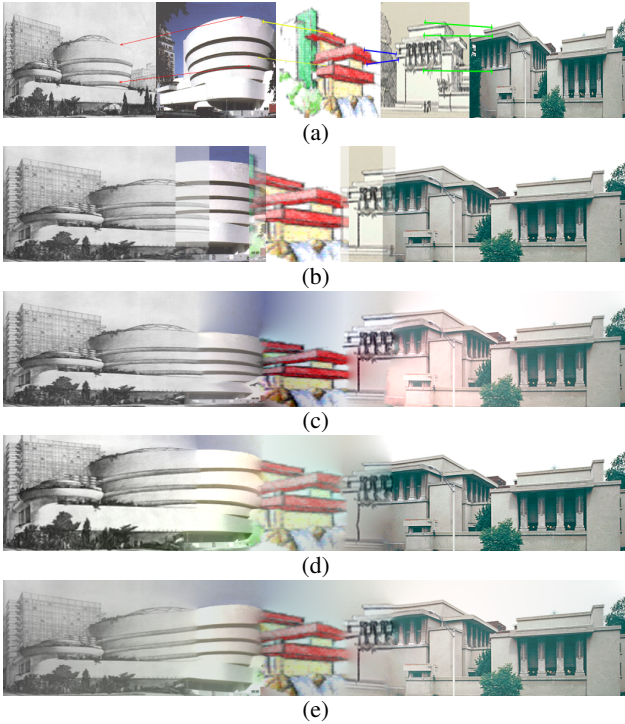


Figure 6. Architecture example. (a) Input images. Note the large discrepancy in structure and color among these images. (b) Roughly aligned images. The overlapping areas are blended. (c) is the stitching result generated by deforming $I_i, i = 2, \dots, n$, to align with the current stitching result $\{I_1, \dots, I_{i-1}\}$, which are fixed during each stitching step. (d) is the stitching result generated by deforming $I_i, i = n - 1, \dots, 1$, to align with the current stitching result $\{I_{i+1}, \dots, I_n\}$ which are fixed in the current step. (e) Our final stitching result shows a better unbiased photomontage where all images are deformed during the stitching process.

deforming I_i to align with the stitched set $\{I_1, \dots, I_{i-1}\}$ while the latter are fixed during each stitching process. Notice the curved ramp is squeezed. Figure 6(d) is analogous to (c) with the stitching direction reversed. Observe the stretched ramp. Figure 6(e) shows the stitching result of an unbiased photomontage where the four pairwise transforms are post-processed successively by enforcing stochastic inverse consistency to avoid biases and accumulation errors.

5. Conclusion

We propose a post-processing algorithm for enforcing stochastic inverse consistency, given a pair of forward and backward fields. By using synthetic data, we showed that the fields processed by our algorithm can satisfy stochastic inverse consistency when applicable. The symmetrized output is unbiased to any input image, allowing the same result (in a stochastic sense) be obtained regardless of the image order. We exploit this advantage and proposed two applications. One application involves symmetrizing match-

ing between a stereo pair, where occlusion boundaries can be localized as violations to inverse consistencies, without any modifications to the chosen matching algorithm. Another application is general image stitching in the presence of severe intensity and structure misalignment among the images. Our result suffers less biases as a result of enforcing stochastic inverse consistency among the input images during the stitching process.

References

- [1] B. B. Avants, M. Grossman, and J. C. Gee. Symmetric diffeomorphic image registration: Evaluating automated labeling of elderly and neurodegenerative cortex and frontal lobe. In *WBIR*, pages 50–57, 2006.
- [2] M. F. Beg and A. Khan. Symmetric data attachment terms for large deformation image registration. *IEEE Trans. Med. Imaging*, 26(9):1179–1189, 2007.
- [3] M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *IJCV*, 61(2):139–157, 2005.
- [4] V. Camion and L. Younes. Geodesic interpolating splines. In *EMMCVPR*, pages 513–527, 2001.
- [5] G. E. Christensen and H. J. Johnson. Consistent image registration. *IEEE Trans. Med. Imaging*, 20(7):568–582, 2001.
- [6] Y. Deng, Q. Yang, X. Lin, and X. Tang. Stereo correspondence with occlusion handling in a symmetric patch-based graph-cuts model. *PAMI*, 29(6):1068–1079, June 2007.
- [7] J. Jia and C.-K. Tang. Eliminating structure and intensity misalignment in image stitching. In *ICCV*, pages 1651–1658, 2005.
- [8] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV (3)*, pages 82–96, 2002.
- [9] A. Leow, S.-C. Huang, A. Geng, J. T. Becker, S. Davis, A. W. Toga, and P. Thompson. Inverse consistent mapping in 3d deformable image registration: Its construction and statistical properties. In *IPMI*, pages 493–503, 2005.
- [10] A. Levin, A. Zomet, S. Peleg, and Y. Weiss. Seamless image stitching in the gradient domain. In *ECCV (4)*, pages 377–389, 2004.
- [11] S. Marsland and C. J. Twining. Constructing diffeomorphic representations for the groupwise analysis of nonrigid registrations of medical images. *IEEE Trans. Med. Imaging*, 23(8):1006–1020, 2004.
- [12] V. Noblet, C. Heinrich, F. Heitz, and J.-P. Armspach. A topology preserving non-rigid registration method using a symmetric similarity function-application to 3-d brain images. In *ECCV (3)*, pages 546–557, 2004.
- [13] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.
- [14] J. Sun, Y. Li, S. Kang, and H. Shum. Symmetric stereo matching for occlusion handling. In *CVPR05*, pages II: 399–406, 2005.
- [15] H. Tagare, D. Groisser, and O. Skrinjar. A geometric theory of symmetric registration. In *MMBIA06*, 2006.

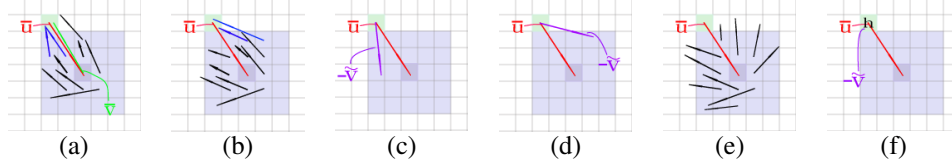


Figure 7. Computing the inverse map $\tilde{\mathbf{V}}$. Given a vector $\bar{\mathbf{u}}$, the corresponding inverse vector $\bar{\mathbf{v}}$ is found in the search region (the pale blue area) within $\tilde{\mathbf{V}}$, the size of this region depends on the magnitude of $\bar{\mathbf{u}}$. (a) There is a green $\bar{\mathbf{v}}$ and two more vectors within the search region pointing toward the location that contains $\bar{\mathbf{u}}$. (b) The direct inverse green $\bar{\mathbf{v}}$ is not found, so the corresponding inverse vector is formed by the blue vectors. (c)–(d) The purple vector $-\tilde{\mathbf{v}}$ is the average of all the inverse vector candidates. (e) There may be situation where no $\bar{\mathbf{v}}$ can be found as an inverse candidate of $\bar{\mathbf{u}}$, in such case a ‘hole’ will be present in $\tilde{\mathbf{V}}$, as shown in (f). Similar procedure for establishing $\tilde{\mathbf{U}}$. [This is a color-coded figure.]

- [16] S. Van Huffel and J. Vandewalle. Analysis and properties of the generalized total least squares problem $AX \approx B$ when some or all columns in A are subject to error. *SIAM J. Matrix. Anal. Appl.*, 10:294–315, 1989.
- [17] T.-P. Wu, K.-L. Tang, C.-K. Tang, and T.-T. Wong. Dense photometric stereo: A markov random field approach. *PAMI*, 28(11):1830–1846, November 2006.
- [18] S.-K. Yeung and P. Shi. Stochastic inverse consistency in medical image registration. In *MICCAI*, pages 234–241, 2005.
- [19] K. Yoon and I. Kweon. Stereo matching with symmetric cost functions. In *CVPR06*, pages II: 2371–2377, 2006.

A. TBP algorithm

1. Initialize $m_{st}(s, t)$ as a $n \times n$ identity matrix, and $m_s(s) = \bar{\mathbf{u}}_o(s)\bar{\mathbf{u}}_o(s)^T$ to indicate the initial belief in the vector for site s , where $\bar{\mathbf{u}}_o(s) \in \tilde{\mathbf{U}}$.
 2. Update messages $m_{st}(s, t)$ iteratively for N iterations (N is set according to the size of input image):
- 2.1 Find the current vector with the highest probability:

$$b_s(s) \leftarrow m_s(s) + \sum_{k \in \mathbf{N}(s)} m_{ks}(k, s) \quad (10)$$

$$\bar{\mathbf{u}}(s) \leftarrow \hat{e}_1[b_s(s)] \quad (11)$$

where $\mathbf{N}(s)$ is the first-order neighborhood of s , and $\hat{e}_1[b_s(s)]$ is the unit eigenvector associated with the largest eigenvalue of the tensor $b_s(s)$.

- 2.2 Compute new messages:

$$m_{st}(s, t) \leftarrow \varphi_{st}(\bar{\mathbf{u}}(s), \bar{\mathbf{u}}(t)) (\text{normalize}[m_s(s) + \sum_{k \in \mathbf{N}(s) \setminus t} m_{ks}(k, s)]) \quad (12)$$

where the normalization scales all eigenvalues so that the largest one equals to 1 and $\varphi_{st}(\bar{\mathbf{u}}(s), \bar{\mathbf{u}}(t)) = \exp(-\frac{\log(1 + \frac{1}{2}(\frac{\|\bar{\mathbf{u}}(s) - \bar{\mathbf{u}}(t)\|}{\sigma})^2)}{2\sigma^2})$ is defined by the robust estimator (Lorentzian) to preserve the discontinuity between s and t .

3. Compute beliefs:

$$b_s(s) \leftarrow m_s(s) + \sum_{k \in \mathbf{N}(s)} m_{ks}(k, s) \quad (13)$$

$$\bar{\mathbf{u}}(s) \leftarrow \hat{e}_1[b_s(s)] \quad (14)$$

B. Computing inverse map

See Figure 7.

C. GTLS

We summarize GTLS [16] here. Consider an overdetermined system of linear equations with a set of m linear equations in $n \times d$ unknowns X :

$$AX \approx B \quad A \in R^{m \times n}, B \in R^{m \times d} \text{ and} \\ X \in R^{n \times d}, m \geq n + d \quad (15)$$

Partition $A = [A_1; A_2]$, where $A_1 \in R^{m \times n_1}, A_2 \in R^{m \times n_2}$ and $n = n_1 + n_2$, and also partition $X = [X_1^T; X_2^T]^T$, where $X_1 \in R^{n_1 \times d}$ and $X_2 \in R^{n_2 \times d}$. Assume that the columns of A_1 are error free and that non-singular error equilibration matrices $R_D \in R^{m \times m}$ and $R_C \in R^{(n_2+d) \times (n_2+d)}$ are given such that the errors on $R_D^{-T}[A_2, B]R_C^{-1}$ are equilibrated, that is, uncorrelated with zero mean and same variance. Then, a GTLS solution of Eq. (15) is any solution of the set $\hat{A}X = A_1X_1 + \hat{A}_2X_2 = \hat{B}$ where $\hat{A} = [A_1, \hat{A}_2]$ and \hat{B} are determined such that

$$\text{Range}(\hat{B}) \subseteq \text{Range}(\hat{A}) \quad (16)$$

and

$$\|R_D^{-T}[\Delta \hat{A}_2, \Delta \hat{B}]R_C^{-1}\|_F \\ = \|R_D^{-T}[A_2 - \hat{A}_2, B - \hat{B}]R_C^{-1}\|_F \quad (17)$$

is minimal. The problem of finding $[\Delta \hat{A}_2, \Delta \hat{B}]$ such that Eqs (16) and (17) are satisfied is referred to as the GTLS problem. Whenever the solution is not unique, GTLS singles out the minimum norm solution, denoted by $\hat{X} = [\hat{X}_1^T; \hat{X}_2^T]^T$.