

Photographic stitching with optimized object and color matching based on image derivatives

Simon T.Y. Suen, Edmund Y. Lam, and Kenneth K.Y. Wong

Department of Electrical and Electronic Engineering,
The University of Hong Kong,
Pokfulam Road,
Hong Kong

{tysuen,elam,kywong}@eee.hku.hk

Abstract: In this paper, a novel optimization-based stitching method is presented. It minimizes an energy function defined with derivatives up to the second order. We have identified some appropriate choices for its parameters, allowing it to reduce artifacts such as ghosting, color inconsistency, and misalignment. To accelerate the computation, a multi-resolution technique is introduced. The significant speedup and memory saving make it possible for use in hand-held capturing devices.

© 2007 Optical Society of America

OCIS codes: (100.2000) Digital image processing; (100.3020) Image reconstruction-restoration; (110.5200) Photography.

References and links

1. F. P. Araújo Jr. and N. J. Leite, "A morphological algorithm for photomosaicking," in *Proceedings of 8th European Signal Processing Conference (EURASIP, 1996)*, pp. 1881–1884.
2. F. Meyer and S. Beucher, "Morphological segmentation," *J. Vis. Comm. and Imag. Rep.* **1**, 21–46 (1990).
3. M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating ghosting and exposure artifacts in image mosaics," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (IEEE, 2001)*, pp. 509–516.
4. P. Soille, "Morphological image compositing," *IEEE Trans. Pat. Anal. and Mach. Intell.* **28**, 673–683 (2006).
5. A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *Proceedings of 8th European Conference on Computer Vision (2004)*, pp. 377–389.
6. C. T. Hsu and J. L. Wu, "Multiresolution mosaic," *IEEE Trans. Cons. Elec.* **42**, 981–990 (1996).
7. M. S. Su, W. L. Hwang, and K. Y. Cheng, "Variational calculus approach to multiresolution image mosaic," in *Proceedings of International Conference on Image Processing (IEEE, 2001)*, pp. 245–245.
8. M. S. Su, W. L. Hwang, and K. Y. Cheng, "Analysis on multiresolution mosaic images," *IEEE Trans. Imag. Proc.* **13**, 952–959 (2004).
9. A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of SIGGRAPH '01: 28th Annual Conference on Computer Graphics and Interactive Techniques (ACM, 2001)*, pp. 341–346.
10. R. Gonzalez and R. Woods, *Digital Image Processing* (NJ: Prentice Hall, 2002).
11. S. T. Suen, E. Y. Lam, and K. K. Wong, "Digital photograph stitching with optimized matching of gradient and curvature," *Proc. SPIE* **6069**, 139–154 (2006).
12. S. Boyd and L. Vandenberghe, *Convex Optimization* (UK: Cambridge University Press, 2004).
13. S. T. Suen, E. Y. Lam, and K. K. Wong, "Photographic mosaic for camera phones based on minimization of curvature value variations," Tech. rep., Department of Electrical and Electronic Engineering, The University of Hong Kong (2006), http://www.eee.hku.hk/research/research_reports.htm.
14. G. Strang and T. Nguyen, *Wavelets and Filter Banks* (MA: Wellesley-Cambridge Press, 1996).
15. "The Panorama Factory V4.4 for Windows XP," <http://www.panoramafactory.com/>.

1. Introduction

Photographic stitching is a process that matches the color of multiple photographs and blends them into one. Artifacts may occur when there are misalignments in spatial position or color. When objects move between different captures, a conspicuous object cut appears if we cannot locate the cut around the affected regions. In addition, color inconsistency due to variations in lighting or internal camera setting would introduce a sharp transition.

We can classify the existing stitching methods as follows. Araújo and Leite [1] define a mask image representing the similarity of the images' intensities, and then creates an optimal irregular cut by watershed segmentation [2]. Concatenation of images at this cut can be seamless only when color inconsistency does not appear throughout the overlap. Moreover, cutting of moving objects can be prevented if the color is compensated before forming the mask image, and the objects' intensities are not similar to the background. Another intensity method, Exposure Compensation [3] spreads the inconsistency to non-overlapping regions iteratively by estimating and averaging the photometric properties among the images in each step. However, this is subjected to double-edge artifacts for slightly misaligned images. Soille [4] searches on a mask image defined with consistent gradients, and using watershed segmentation it attempts to minimize the cut along salient edges in the image. Moving objects may be cut through when the edges are not strong, or consistent edges appear in front of the objects. Gradient Stitching [5] optimizes the matching of first derivatives of the image intensities, suppressing the creation of sharp transition in the flat region of the overlap and producing a result which is less sensitive to average intensity differences. However, sharp transition appears at some regions when the inconsistency is not uniform. Wavelet Blending [6, 7, 8] mixes the subband images at each level within a transition whose width is comparable to the object feature size. The reconstructed image can prevent double edges and give a gradual transition. However, it is not easy for users to control the behavior of the transition.

In this paper, we propose to optimize the matching of second derivatives in the transition region in order to create a smooth transition even for non-uniform inconsistency. In addition, we develop a morphological method to search for an optimal cut without passing through moving objects by using intensity and gradient similarity. The cut is converted into a variable in our optimization function to prevent ghosting and double edges. A multi-resolution technique is applied to process large digital photographs.

2. Our optimization stitching method

We define an energy function $\Psi\{\hat{g}(x,y)\}$ of the mosaic result $\hat{g}(x,y)$, which includes derivatives up to the second order, where

$$\Psi\{\hat{g}(x,y)\} = \sum_{i=0}^2 \left(\sum_{(x,y) \in \Omega_1} \alpha(x,y) \beta_i(x,y) |D^i(\hat{g}(x,y) - g_1(x,y))| + \sum_{(x,y) \in \Omega_2} (1 - \alpha(x,y)) \beta_i(x,y) |D^i(\hat{g}(x,y) - g_2(x,y))| \right). \quad (1)$$

We seek to minimize this function by choosing $\hat{g}(x,y)$ for every pixel location. Here, Ω_1 and Ω_2 are the capturing regions of the input images $g_1(x,y)$ and $g_2(x,y)$ respectively. $D^i f$ is the i^{th} derivative of the real function f such that $D^0 f = f$, $D^1 f = [\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}]$ and $D^2 f = [\frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial y^2}]$. $\alpha(x,y)$ is a binary mask indicating if $g_1(x,y)$ or $g_2(x,y)$ is selected. $\beta_i(x,y)$ is also a binary mask representing whether the i^{th} derivative is selected at pixel (x,y) . In Section 2.1, we explain how to find a curve to avoid passing through inconsistent objects and use it to design $\alpha(x,y)$. In Section 2.2, we will describe the properties of the derivatives and our design of $\beta_i(x,y)$.

2.1. Optimal cut based on intensities and gradients

The binary mask $\alpha(x,y)$ equals 1 if we want to select a pixel from $g_1(x,y)$ and 0 otherwise. As with other pixel selection method [1, 4, 9], we find a suitable curve and assign one side for $g_1(x,y)$. Usually, double-edge artifact can be prevented as no pixels are mixed. Here we introduce a method to give a cut without passing through moving objects. First, we capture the intensity dissimilarity from the overlap region. To compensate for photometric inconsistency, we subtract the mean intensity from each pixel for each input image beforehand. The dissimilarity $a(x,y)$ is represented by their absolute differences:

$$a(x,y) = |g_1(x,y) - g_2(x,y)|. \quad (2)$$

Morphological closing is then applied to connect small gaps in $a(x,y)$. Next, we measure the gradient consistency $b(x,y)$:

$$b(x,y) = \rho_A(g_1(x,y)) \wedge \rho_A(g_2(x,y)), \quad (3)$$

where \wedge denotes the point-wise minimum operator and $\rho_A(\cdot)$ denotes a morphological gradient operator with structuring element A [10]. The size of A increases with more misalignment such that the same edge in $g_1(x,y)$ and $g_2(x,y)$ can manifest as an edge in $b(x,y)$. We combine $a(x,y)$ and $b(x,y)$ to form a mask image $c(x,y)$:

$$c(x,y) = \frac{b(x,y)}{a(x,y)}. \quad (4)$$

By marker-controlled watershed segmentation [2], we immerse corresponding markers from both sides of $c(x,y)$ to form an image with labels for selecting the input images. In theory, the optimal cut is located along consistent edges that are not on moving objects. We then assign $\alpha(x,y) = 1$ to the side of $g_1(x,y)$ and $\alpha(x,y) = 0$ otherwise. For color images, we can choose any one of the color planes to compute the cut and assign its corresponding mask $\alpha(x,y)$ for all the planes. In Sect. 4, our experiment shows that this algorithm works well even on the plane with severe photometric inconsistency.

2.2. Transition smoothness and fidelity

The binary mask $\beta_i(x,y)$ determines the contribution of the derivatives. The zeroth derivatives (i.e. image intensities) contain the complete information of an image and do not reject inconsistency. The first derivatives remove the mean intensities of an image, removing uniform inconsistency because the input images can match each other regardless of their global differences. For the second derivatives, both mean intensities and gradients are removed from an image. Hence, spatially varying inconsistency can also be reduced when images are matched in this domain. Readers can refer to our earlier work [11, 13] for a detailed mathematical analysis.

Suppose non-uniform inconsistency is severe among the input images. Thus, we should emphasize the second derivatives in the transition region. By default, it is the overlap region. However, users can extend it to the non-overlapping regions if a smoother transition is desired. We assign $\beta_2(x,y) = 1$ in the region and 0 elsewhere. In addition, users can assign either the zeroth or the first derivatives to the remaining region according to their requirement on fidelity relative to the input images. If matching the mean intensities is preferred, the first derivatives should be selected. We assign $\beta_1(x,y) = 1$ in the region and 0 elsewhere, while setting $\beta_0(x,y) = 0$ for all pixels. Similarly, if they prefer to keep the intensities, we swap $\beta_0(x,y)$ and $\beta_1(x,y)$ in this assignment rule. These features favor the subjective requirement of the users.

3. Basic implementation and multi-resolution enhancement technique

In our basic implementation, we use a forward differencing derivative filter $[-1 \ 1]$ as the first derivative operator and $[1 \ -2 \ 1]$ as the second derivative operator. The energy function can therefore be converted into a linear programming (LP) according to [11]. Since this is an unconstrained minimization, we can begin the process with any initial estimate of $\hat{g}(x,y)$ denoted as $\hat{g}_0(x,y)$, such as when we simply merge $g_1(x,y)$ and $g_2(x,y)$ according to the pixel locations with respect to the cut, i.e.

$$\hat{g}_0(x,y) = \alpha(x,y)g_1(x,y) + (1 - \alpha(x,y))g_2(x,y). \quad (5)$$

In our implementation, we use `linprog` in Matlab, which employs an interior point algorithm [12].

Furthermore, in order to increase the computational speed, we reduce the number of optimization variables using a multi-resolution technique. Large digital photographs are first decomposed into several resolution levels by 9-7 wavelet transform [14]. The subbands in each level are simply selected to form a predefined curve, except the approximation subbands which are instead stitched by our method. From our experience, the size of the approximation images should be at least 200 pixels to avoid artifacts.

4. Experimental results and discussion

In our experiment, we capture two digital photographs (960×1280 pixels each) at shifted orientation with different white balance settings. Note that some objects move between the two captures and photometric inconsistency exists mainly in green. Before our stitching procedure, the photographs are preprocessed by a commercial software [15]. It automatically warps the images to correct for lens and perspective distortions, and also performs global alignment. The warped images are shown in Fig. 1 and the rectangular box roughly indicates their overlap region.

4.1. Comparisons on identifying an optimal cut

First we compare the ability of finding a cut without passing through moving objects. The curve searching methods are applied in the green plane only to show the superior performance of our technique for images under heavy photometric inconsistency. Araújo and Leite [1] and Soille [4] measure consistency in intensity and gradient respectively, followed by searching a cut using watershed segmentation. Our method instead measures both. The results are shown in Fig. 2, where black lines indicate the cutting curves. In (a), the method of Araújo and Leite locates the curve at the roof of the black van, giving a noticeable cutting. In (b), Soille's method locates the curve along the window of another van, where the cutting is again visible. However, in Fig. 3(a), our method prevents the curve from passing through the moving objects and minimizes the cut along salient edges. As shown in Fig. 3(b), the binary mask $\alpha(x,y)$, where black is 0 and white is 1, can be created according to this curve.

4.2. Comparisons on stitching methods

We compare with other stitching methods both subjectively and objectively, using $\alpha(x,y)$ created above to eliminate ghosting. Also, we apply the multi-resolution enhancement technique for Wavelet Blending, Gradient Stitching and our method. It decomposes the images into 6 wavelet levels and thus the approximation image is 30×40 pixels each. For our method, the optimization process can finish in about 10 seconds for each channel.

We quantify the transition smoothness and fidelity relative to the input images by Mean Square Error (MSE). The smoothness metric M_s is the MSE between the result and the two



Fig. 1. Input photographs with moving objects warped by commercial software. The rectangular boxes roughly indicate their overlap region.



Fig. 2. (a) (b) Results of pixel selection according to the curve generated by Araújo and Leite, and Soille, respectively. The cutting curves are indicated by black lines. The rectangular boxes mark the regions where the curve passes through a moving object.



Fig. 3. (a) Result of pixel selection according to the curve generated by our method. No moving object is cut. (b) Binary mask $\alpha(x,y)$ corresponds to the curve.

input images within the overlap. Smaller M_s implies that the result is more similar to the average of the images, and thus the transition is smoother. The fidelity metric M_f is the MSE outside the overlap regions. A smaller M_f means better fidelity.

Table 1 shows M_s and M_f for (a) Exposure Compensation, (b) Wavelet Blending, (c) Gradient Stitching, and (d) and (e) our method with different $\beta_i(x,y)$. In (d), we assign 1 to the middle 65% vertical portion of $\beta_2(x,y)$, $\beta_0(x,y) = 1$ in the remaining area, and $\beta_1(x,y) = 0$ for all pixels. In (e), we assign 1 to the 85% portion of $\beta_2(x,y)$ instead. Note that since the vehicles shown in Fig. 1(b) are removed in most of the results, we do not count the errors in M_s for those regions. In the Table, Exposure Compensation gives the smallest M_s , followed by our method with the two settings. Thus, our technique gives a smoother result than Wavelet Blending and

Gradient Stitching. Moreover, the result with our second parameter setting is smoother than the first one. Our first design achieves the smallest M_f , followed by our second one. Therefore, our method can maintain the content better than the other methods.

Table 1. Summary of the smoothness metric (M_s) and fidelity metric (M_f) for the stitching methods: (a) Exposure Compensation, (b) Wavelet Blending, (c) Gradient Stitching, (d) Our method minimizing second derivatives over the middle 65% vertical portion, and (e) minimizing over the middle 85% portion.

	(a)	(b)	(c)	(d)	(e)
M_s	636.4725	934.3102	933.1125	932.3330	932.2184
M_f	1.5877	2.3191	42.0599	0.5942	1.2154

Next, we compare their visual results as shown in Fig. 4. In (a), Exposure Compensation gives a very smooth transition at the expense of severe double-edge artifacts. (b) shows that Wavelet Blending gives a noticeable sharp transition across the cutting curve. In (c), Gradient Stitching is much smoother, but mean intensities on the non-overlapping regions are modified heavily. (d) shows that our method with the second parameter setting gives no double edges or noticeable cut. Also, the non-overlapping regions are not significantly altered. When we zoom in the fence, as shown in Fig. 5, we can see that our method gives the smoothest transition across the cutting curve when compared with Wavelet Blending and Gradient Stitching.



Fig. 4. Stitching Result of (a) Exposure Compensation, (b) Wavelet Blending, (c) Gradient Stitching and (d) Our method, assigning 1 to the middle 85% vertical portion of $\beta_2(x, y)$.

Furthermore, another stitching example can be found in [13]. The two photographs shown in Fig. 6, again with different white balance conditions, are stitched. Here, in Tbl. 2, we show the smoothness metric (M_s) and fidelity metric (M_f) after stitching. We can observe that the comparison can match with that presented in this paper.

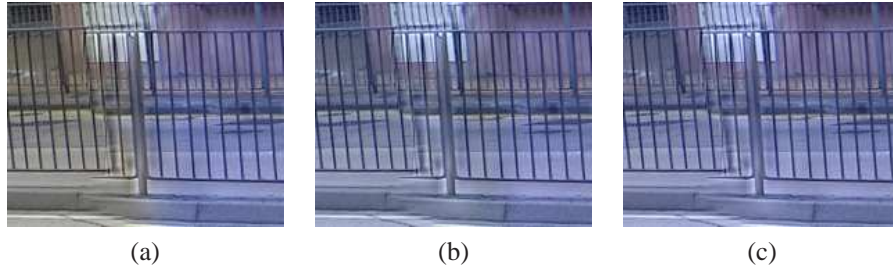


Fig. 5. A zoom-in region of the result (a) Wavelet Blending, (b) Gradient Stitching and (c) Our method. They illustrate the transition across the cutting curve.



Fig. 6. (a) and (b) Input photographs for the stitching example described in [13].

Table 2. Summary of the smoothness metric (M_s) and fidelity metric (M_f) for the stitching methods: (a) Exposure Compensation, (b) Wavelet Blending, (c) Gradient Stitching, (d) Our method minimizing second derivatives over the overlap region and (e) Minimizing throughout the mosaic.

	(a)	(b)	(c)	(d)	(e)
M_s	995	1463	1355	1222	1103
M_f	127	124	862	124	185

4.3. Comparisons on decomposition level count

Finally, we test how the number of decomposition levels in our multi-resolution enhancement technique affects memory usage and running time. We redo the stitching case in Tbl. 1(d) with different number of decomposition levels. In Tbl. 3, the memory usage, total running time and the ratio of time consumed by optimization to the total running time for each color plane on average are tabulated. When there are fewer than 4 levels, our system runs out of memory. However, when it increases, both the total running time and memory decrease, illustrating that this technique can improve our method's efficiency when the number of levels increases. Furthermore, the ratio of time consumption drops with the increase in level count. This suggests that the bottleneck of our method is in the decomposition and reconstruction instead of in the optimization process.

Table 3. Summary of the average memory usage (M), total running time (T_t) and the ratio of time consumed by optimization (T_o) to the total running time when stitching with different number of decomposition levels.

	≤ 3 levels	4 levels	5 levels	6 levels
M	n/a	221 MB	209 MB	159 MB
T_t	n/a	152.7 sec	27 sec	10.6 sec
$T_o : T_t$	n/a	0.91	0.75	0.38

5. Conclusion

In this paper, we propose an optimization method based on image derivatives to stitch photographs with color and position mismatch. To prevent cutting through moving objects, we use a curve based on both intensity and gradient consistency. Then, we blend the colors along the curve by minimizing the dissimilarity of second derivatives between the result and the input photographs. We find that double edges, ghosting and visible cut can be reduced effectively. Moreover, by adjusting the minimization area of the derivatives, we can control the transition smoothness and fidelity of the result. Furthermore, the multi-resolution technique can reduce the process time to a few seconds for each channel independent of the original image size.