Multispectral Image Dense Matching

Xiaoyong Shen[§]   Li Xu[†]   Qi Zhang[§]   Jiaya Jia[§]

[§] The Chinese University of Hong Kong
[†] Image & Visual Computing Lab, Lenovo R&T

1   Multispectral Dense Matching Dataset

We build a dataset including four kinds of multispectral images with labeled key point correspondence. The dataset includes 7 RGB/NIR image pairs, 4 RGB/Depth image pairs, 3 Flash/no-flash image pairs and 4 different exposure image pairs. In each pair, we uniformly select corner points and label their correspondences. The resolution of the RGB/Depth image pairs is 640×480 and all the other image pairs are with size 800×600. Our images are shown below.

1.1   RGB/NIR Images

The RGB and NIR image are captured by RGB and NIR cameras respectively. Figs. 1 and 2 show our data, including indoor and outdoor scenes. All image pairs are with rigid and nonrigid transform. The outdoor scene images are obtained from [1].

1.2   RGB/Depth Images

We get the RGB and depth image pairs by a Microsoft Kinect. Our images are shown in Fig. 3. To handle holes and noise in the depth image before matching, bilateral filter [6] is applied.

1.3   Flash/No-flash Images

The flash/no-flash images are captured in dim light. The no-flash images are very noisy since it is produced with the high ISO setting. The flash images are with shadow. Fig. 7 shows the images.
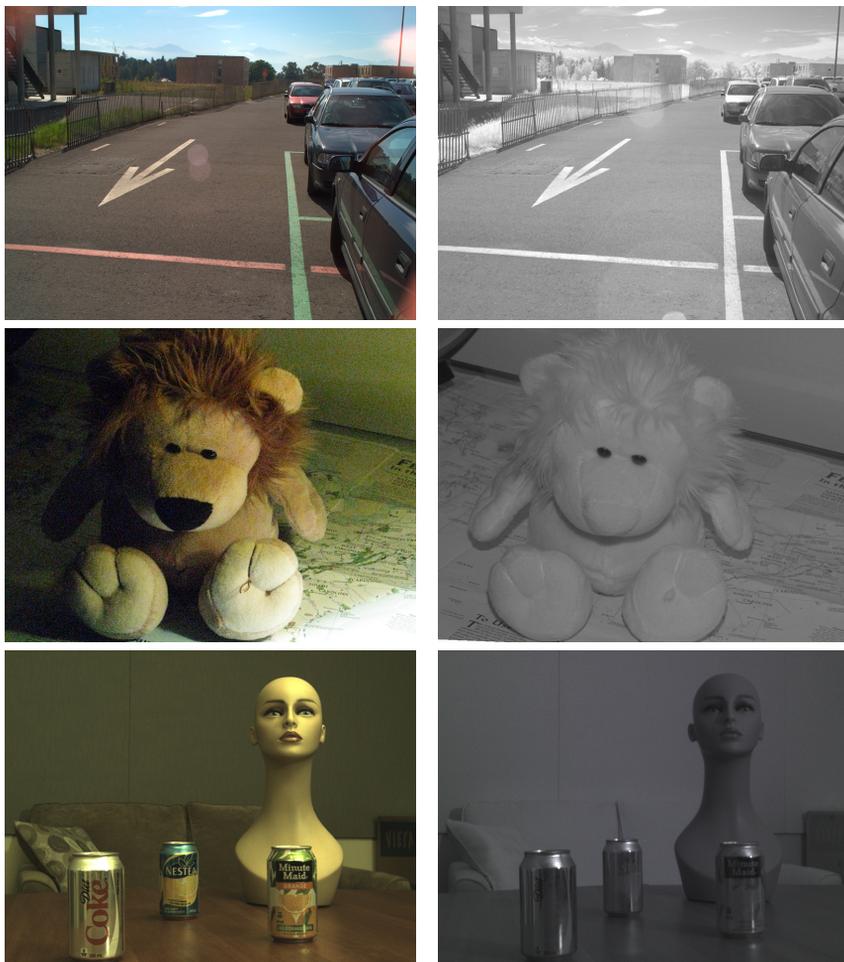
1.4   Different Exposure Images

The different exposure images are obtained from [4]. We crop the image into 800×600 resolution containing the significant dynamic object moving part. The images are shown in Fig. 5.
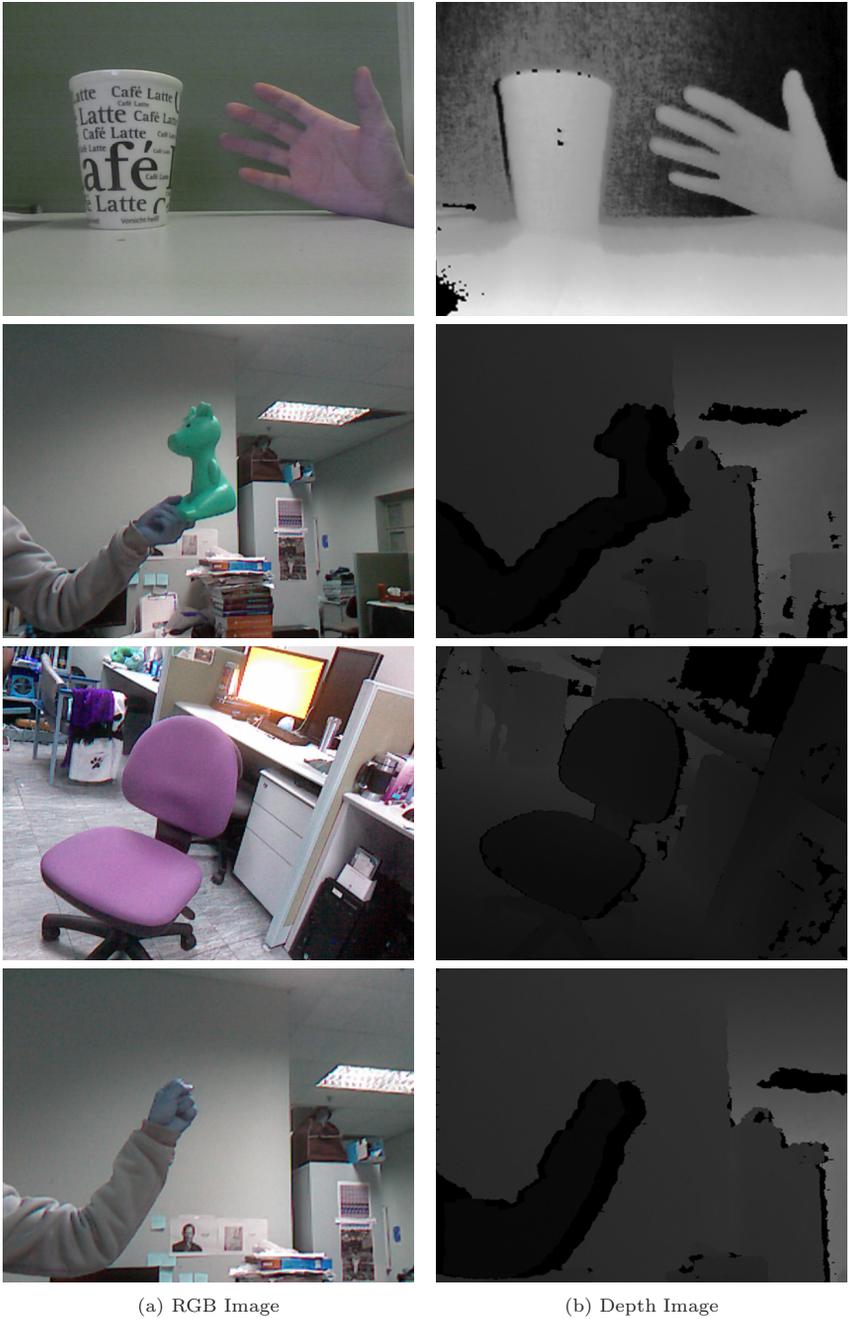
(a) RGB Image   (b) NIR Image

Fig. 1. RGB/NIR image pairs.

(a) RGB Image   (b) NIR Image

Fig. 2. RGB/NIR image pairs.

(a) RGB Image   (b) Depth Image
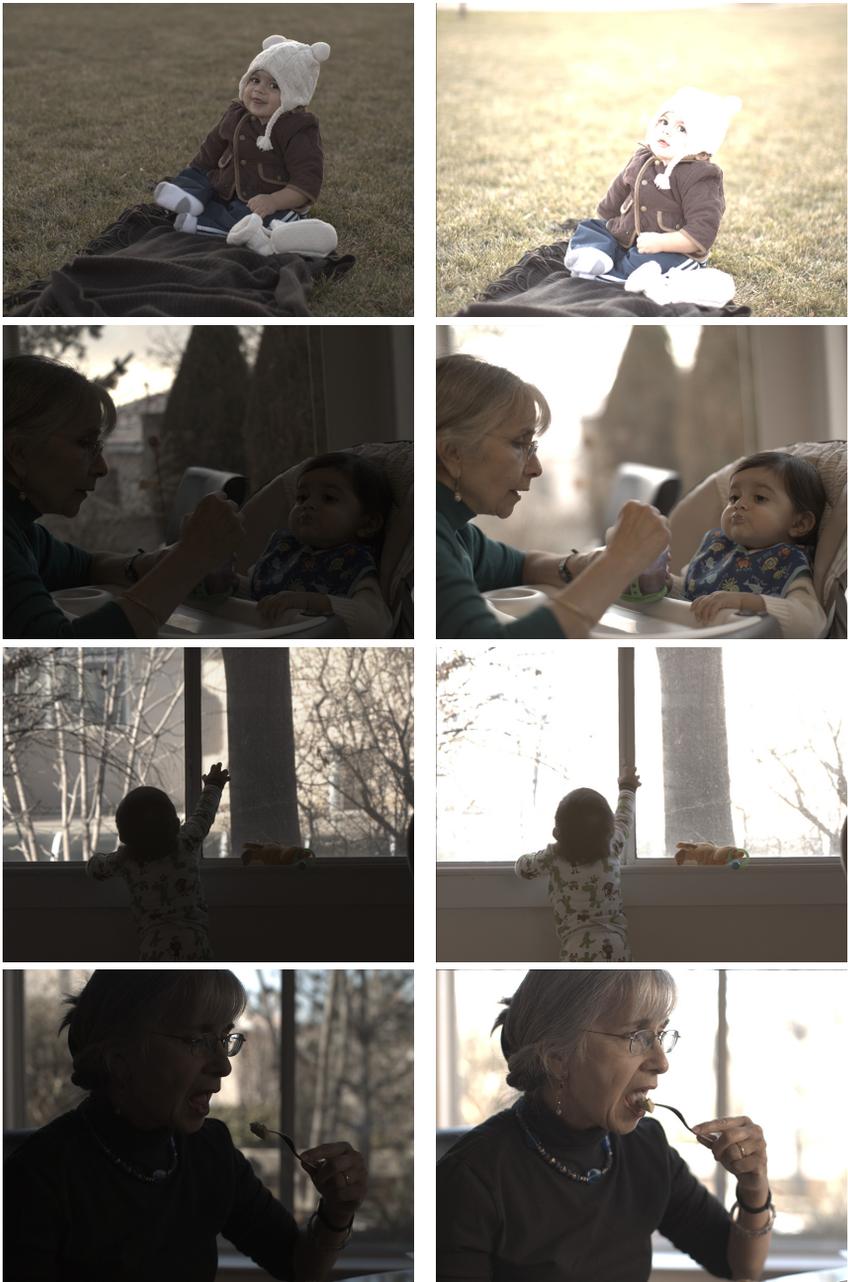
Fig. 3. RGB/Depth image pairs.

(a) Flash Image   (b) No-flash Image

Fig. 4. Flash/no-flash image pairs.

(a) Different Exposures Image 1   (b) Different Exposures Image 2

Fig. 5. Different exposures time image pairs.

2   Local Dense Matching Optimization

As discussed in the paper, our local dense matching framework is defined as

$$E(\mathbf{w}) = \sum_p E^{\text{RSNCC}}(p, w_p) + \lambda_1 \sum_p \psi(\|\nabla w_p\|^2) + \lambda_2 \sum_{q \in N(p)} \|w_p - w_q\|, \quad (1)$$

where $\mathbf{w} = (\mathbf{u}^T, \mathbf{v}^T)^T$ is the vector form of w_p and $\mathbf{u}$. $\mathbf{v}$ are vectors of u_p and v_p respectively. For simplicity's sake, we denote the three terms as $E_D(\mathbf{w})$, $E_S(\mathbf{w})$, and $E_{NL}(\mathbf{w})$. λ_1 and λ_2 are two parameters.

We perform the optimization scheme in a course-to-fine manner for high accuracy during optimizing $E(\mathbf{w})$. In each level, $E(\mathbf{w})$ is updated and then is propagated to the next level as initialization. The process is depicted in Alg. 1.

To handle the non-convex $E(\mathbf{w})$ in each level, we decompose the energy function $E(\mathbf{w})$ into two sub-functions that both find optimal solutions with the scheme of variable-splitting [7]. The two functions are

$$E(\mathbf{w}, \widehat{\mathbf{w}}) = E_D(\mathbf{w}) + \lambda_1 E_S(\mathbf{w}) + \frac{1}{\theta}\|\mathbf{w} - \widehat{\mathbf{w}}\|^2, \quad (2)$$

$$E(\widehat{\mathbf{w}}, \mathbf{w}) = \frac{1}{\theta}\|\widehat{\mathbf{w}} - \mathbf{w}\|^2 + \lambda_2 E_{NL}(\widehat{\mathbf{w}}), \quad (3)$$

where $\widehat{\mathbf{w}}$ is an auxiliary variable. When $\theta \to 0$, the decomposition approaches the original $E(\mathbf{w})$ with high quality.

Our method minimizes Eqs. (2) and (3) respectively. The minimum of Eq. (3) can be obtained by the method of [5]. We now describe how to optimize Eq. (2) efficiently.

We solve Eq. (2) based on the variational configuration using reweighted least squares. In each step, we update the result by a small $\delta\mathbf{w}$ after optimizing $E(\mathbf{w} + \delta\mathbf{w}, \widehat{\mathbf{w}})$. It is done by setting $\frac{\partial E(\mathbf{w}+\delta\mathbf{w}, \widehat{\mathbf{w}})}{\partial \delta\mathbf{w}} = 0$, yielding

$$\begin{bmatrix} \mathbf{B}_x + \lambda_1 \mathbf{L} + \frac{1}{\theta}\mathbf{I} & 0 \\ 0 & \mathbf{B}_y + \lambda_1 \mathbf{L} + \frac{1}{\theta}\mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix} = - \begin{bmatrix} \mathbf{A}_x + \lambda_1 \mathbf{L}\mathbf{u} + \frac{1}{\theta}\widehat{\mathbf{u}} \\ \mathbf{A}_y + \lambda_1 \mathbf{L}\mathbf{v} + \frac{1}{\theta}\widehat{\mathbf{v}} \end{bmatrix} \quad (4)$$

where $\delta\mathbf{u}$ and $\delta\mathbf{v}$ are the updating vectors, and $\mathbf{u}$ and $\mathbf{v}$ are the current displacement estimates. $\mathbf{B}_x$, $\mathbf{B}_y$, $\mathbf{A}_x$, and $\mathbf{A}_y$ are diagonal matrices defined as

$$(\mathbf{A}_x)_{pp} = \sum_{q \in N(p)} (\omega_q^I a_{p,q}^{I,x} + \omega_q^{\nabla I} a_{p,q}^{\nabla I,x}), \quad (5)$$

$$(\mathbf{A}_y)_{pp} = \sum_{q \in N(p)} (\omega_q^I a_{p,q}^{I,y} + \omega_q^{\nabla I} a_{p,q}^{\nabla I,y}), \quad (6)$$

$$(\mathbf{B}_x)_{pp} = \sum_{q \in N(p)} (\omega_q^I b_{p,q}^{I,x} + \omega_q^{\nabla I} b_{p,q}^{\nabla I,x}), \quad (7)$$

$$(\mathbf{B}_y)_{pp} = \sum_{q \in N(p)} (\omega_q^I b_{p,q}^{I,y} + \omega_q^{\nabla I} b_{p,q}^{\nabla I,y}), \quad (8)$$

Algorithm 1 Local Dense Matching Optimization

Input:
  Multispectral images I_1 and I_2.
Output:
  Displacement field **w**.
 1: initialize λ_1, λ_2, θ, $\mathbf{w} = \mathbf{0}$.
 2: **for** level = 1 to MAXLEV **do**
 3:    **for** itr = 1 to MAXITR **do**
 4:       Solve Eq. (3).
 5:       Solve Eq. (2).
 6:       $\theta \leftarrow \frac{\theta}{3}$.
 7:       Update $\mathbf{B}_x$, $\mathbf{B}_y$, $\mathbf{A}_x$, and $\mathbf{A}_y$.
 8:    **end for**
 9:    Re-sample **w** to the finer level.
10: **end for**

---

where ω_p^I and $\omega_p^{\nabla I}$ are weights defined in Eq. (8) in the paper. $a_{p,q}^{F,x}$ where $F \in \{I, \nabla I\}$ is the corresponding coefficient of u_q in $\mathbf{A}_p^F$ and $a_{p,q}^{F,y}$ corresponds to v_q. $b_{p,q}^{F,x}$ and $b_{p,q}^{F,y}$ are denoted similarly according to $\mathbf{B}_p^F$. $\mathbf{L}$ is the Laplacian smoothing matrix given by

$$\mathbf{L} = \mathbf{D}_x^T \Psi' \mathbf{D}_x + \mathbf{D}_y^T \Psi' \mathbf{D}_y, \tag{9}$$

where $\mathbf{D}_x$ and $\mathbf{D}_y$ are the discrete backward difference matrices, which are used to compute image gradient in x- and y-directions. Ψ' is a diagonal matrix that defines the robust weight, whose diagonal elements are $(\Psi')_{pp} = \psi'(|\nabla u_p|^2 + |\nabla v_p|^2)$. $\psi'(x)$ here is the differentiation of $\psi(x)$.

3   More Results

As discussed in our paper, the Multispectral dense image matching is with great challenges due to the structure inconsistency. Many applications can be benefitted by our method. We show a few more examples in the file.

  Our method is appropriate for matching of different exposure images that are not well aligned for restoration of high dynamic range images from low dynamic range ones. Fig. 6(a)-(c) are different exposure time images from [4]. Our Multispectral dense matching method maps (a) and (b) to (c) as shown in Fig. 6(d)-(g). After matching the images, we employ the method proposed in [4] to merge them into a high dynamic range image shown in Fig. 6(h). (i) is our tone mapping result.
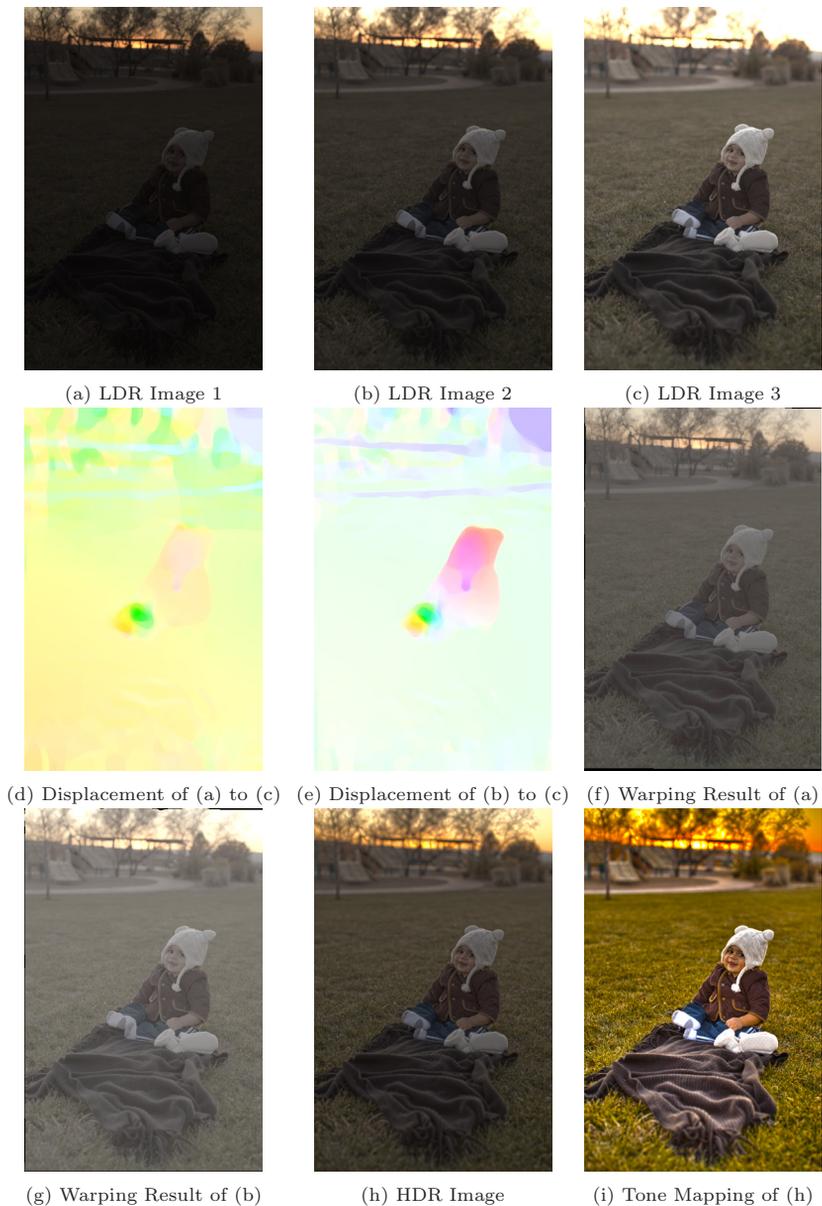
  Additionally, our matching framework can be applied to flash/no-flash image alignment as presented in Figs. 7 and 8. As flash images also suffer from shadow and highlight, our method can work robustly against this kind of outliers as shown in Fig. 7(d). The input images are provided in [3]. Fig. 8 is another example.

We then show an example of depth image and RGB image matching in Fig. 9. Depth image captured by Kinect or other equipments may be different in structure compared with the corresponding RGB image as shown in Fig. 9(c). Moreover, the captured depth image is with noise and holes, which should be removed by smoothing with a kind of filters. However, single-image filtering steps could destroy to an extend original depth structure. Our method matches the smoothed depth image to the corresponding RGB image, and then apply the method of [8] to restore and sharpen edges. Our alignment and restoration results are shown in Fig. 9(d) and (f).
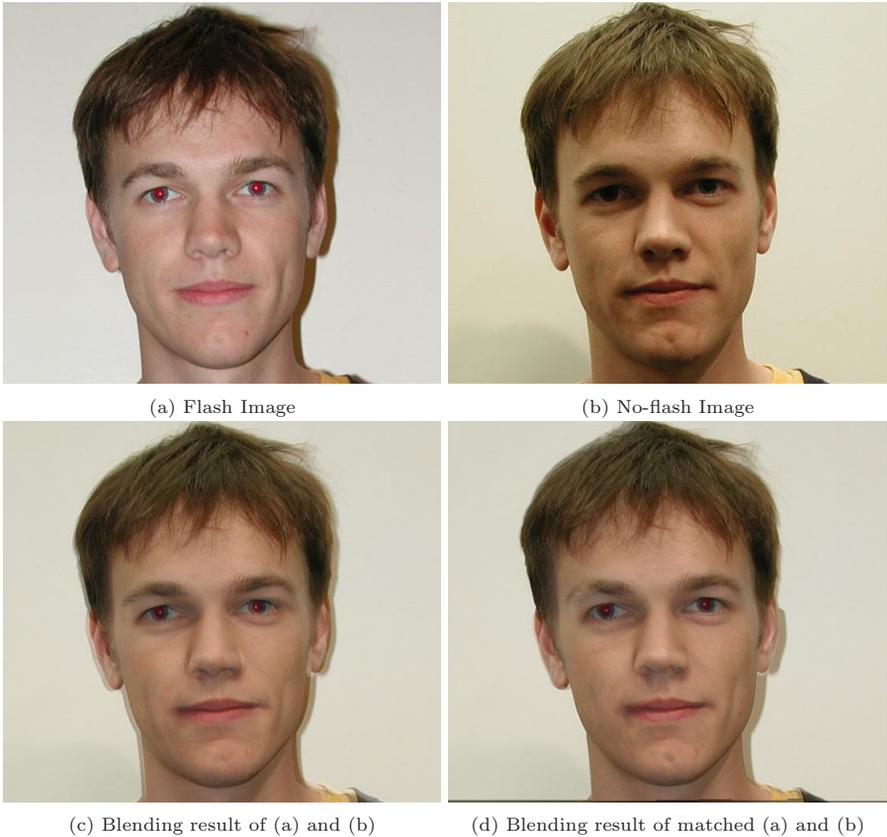
NIR image is a good tool to restore the corresponding noisy RGB image as described in [8]. However, RGB and NIR images are often captured by different cameras and should be aligned before restoration. It is a challenging work due to their nonrigid transform. Our method can well handle nonrigid transform as shown in Figs. 10 and 11. After matching the RGB and NIR images using our method, we perform the Multispectral joint image restoration [8]. The results are shown in Fig. 10(d), Fig. 11(h) and Fig. 12(d). Fig. 13 is another example to handle dense matching of RGB and NIR Multispectral images with gradient reverse.

References

1. Brown, M., Süsstrunk, S.: Multispectral SIFT for scene category recognition. In: CVPR. pp. 177–184 (2011)
2. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.T.: Sift flow: Dense correspondence across different scenes. In: ECCV. pp. 28–42 (2008)
3. Petschnigg, G., Szeliski, R., Agrawala, M., Cohen, M.F., Hoppe, H., Toyama, K.: Digital photography with flash and no-flash image pairs. ToG 23(3), 664–672 (2004)
4. Sen, P., Kalantari, N.K., Yaesoubi, M., Darabi, S., Goldman, D.B., Shechtman, E.: Robust patch-based hdr reconstruction of dynamic scenes. ToG 31(6), 203 (2012)
5. Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: CVPR. pp. 2432–2439 (2010)
6. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: ICCV. pp. 839–846 (1998)
7. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM Journal on Imaging Sciences 1(3), 248–272 (2008)
8. Yan, Q., Shen, X., Xu, L., Zhuo, S., Zhang, X., Shen, L., Jia, J.: Cross-field joint image restoration via scale map. In: ICCV (2013)

(a) LDR Image 1   (b) LDR Image 2   (c) LDR Image 3

(d) Displacement of (a) to (c)   (e) Displacement of (b) to (c)   (f) Warping Result of (a)

(g) Warping Result of (b)   (h) HDR Image   (i) Tone Mapping of (h)

Fig. 6. Low dynamic range (LDR) images to one high dynamic range (HDR) image. (a)-(c) Three LDR images under different exposures and misaligned pixels. (d)-(e) Displacement field from (a) to (c) and (b) to (c) respectively. (f)-(g) Corresponding warping results. (f) and (g) are corrected with gamma = 2.2. (h) Constructed HDR image from (c), (f), and (g). (i) Tone mapping result of (h).

(a) Flash Image

(b) No-flash Image

(c) Blending result of (a) and (b)

(d) Blending result of matched (a) and (b)

Fig. 7. Flash/No-flash matching example. (a)-(b) Unmatched Flash/No-flash images. (c) Blending result of (a) and (b). (d) Result by blending matched (a) and (b). (c) and (d) show misalignment exists in input images and our method effectively eliminates it for further restoration operations. Input images are from [3].

(a) Flash Image

(b) No-flash Image

(c) Blending result of (a) and (b)

(d) Blending result of matched (a) and (b)
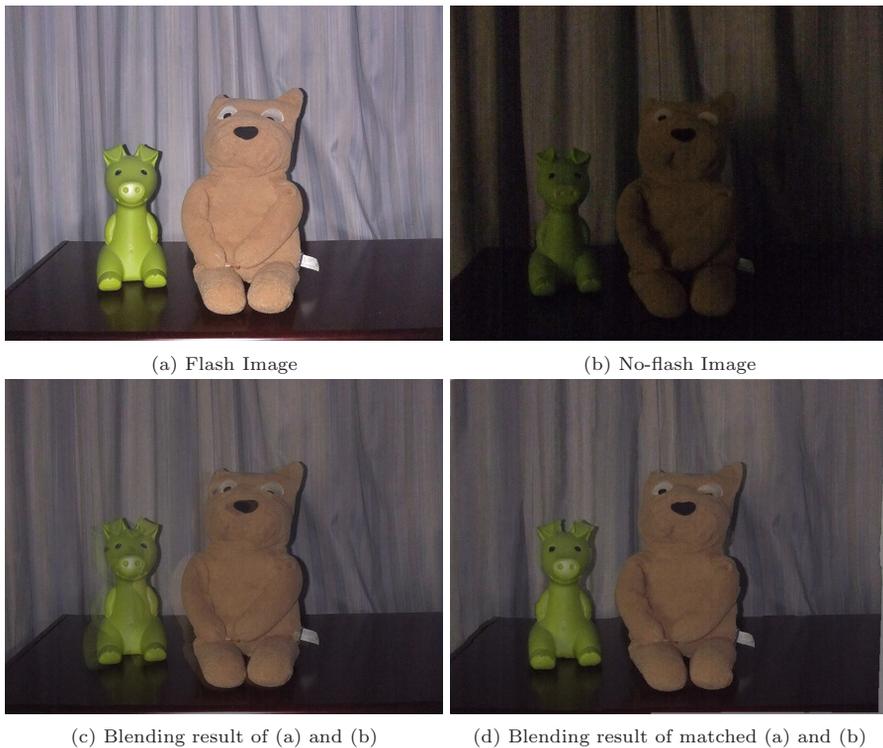
Fig. 8. Flash/No-flash matching example. (a)-(b) Unmatched Flash/No-flash images. (c) Blending result of (a) and (b). (d) Result by blending matched (a) and (b). (c) and (d) show misalignment exists in input images and our method effectively eliminates it for further restoration operations.

(a) RGB Input Image
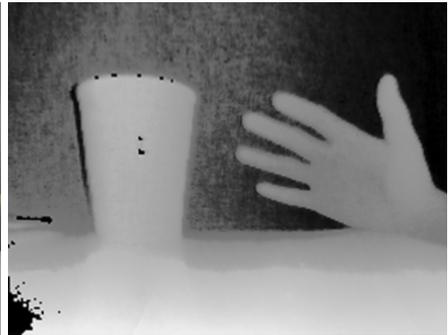
(b) Raw Depth Image

(c) Blending of (a) and (e)

(d) Blending of (a) and Warped (e)

(e) Smoothed Depth Image

(f) Depth Restored with Two Aligned Inputs

Fig. 9. RGB and depth images matching. (a)-(b) are the RGB and depth raw data from [8]. (c)-(d) show the blending results of RGB and depth images and of RGB and aligned depth images. They illustrate how large displacements are. (e) is the single-depth-map smoothed version. (f) is the restoration result based on (a) and matched (e). Result in (f) preserves better structures without fattening artifacts.

(a) Noisy RGB Image   (b) NIR Image

(c) Restored without Matching   (d) Restored with Matching
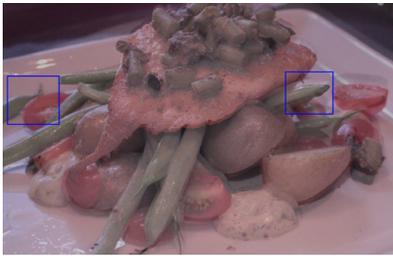
(e) Close-ups

Fig. 10. Multispectral image restoration example. (a) and (b) are unmatched noisy RGB and clean NIR image respectively. (c) is the restoration result without matching (a) and (b) while (d) is the result employing our matching method. The Multispectral image restoration algorithm is our implementation of the one presented in [8]. (e) shows the close-ups.
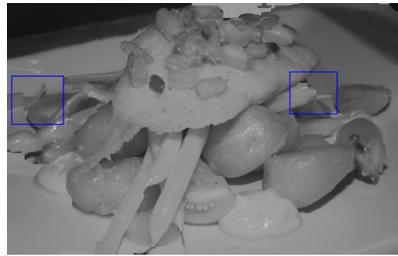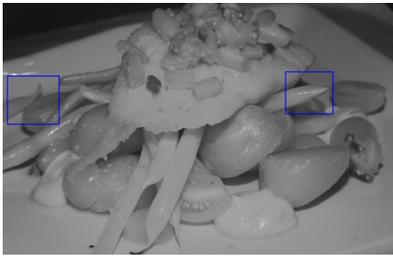
(a) Noisy RGB Image
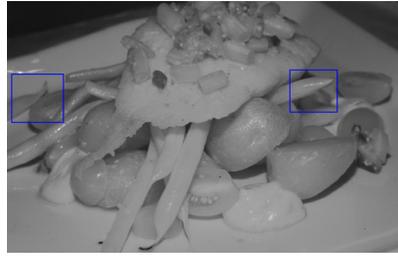
(b) NIR Image

(c) Blending Result of (a) and (b)

(d) Warping Result of SIFT Flow from (b) to (a)

(e) Our Warping Result from (b) to (a)
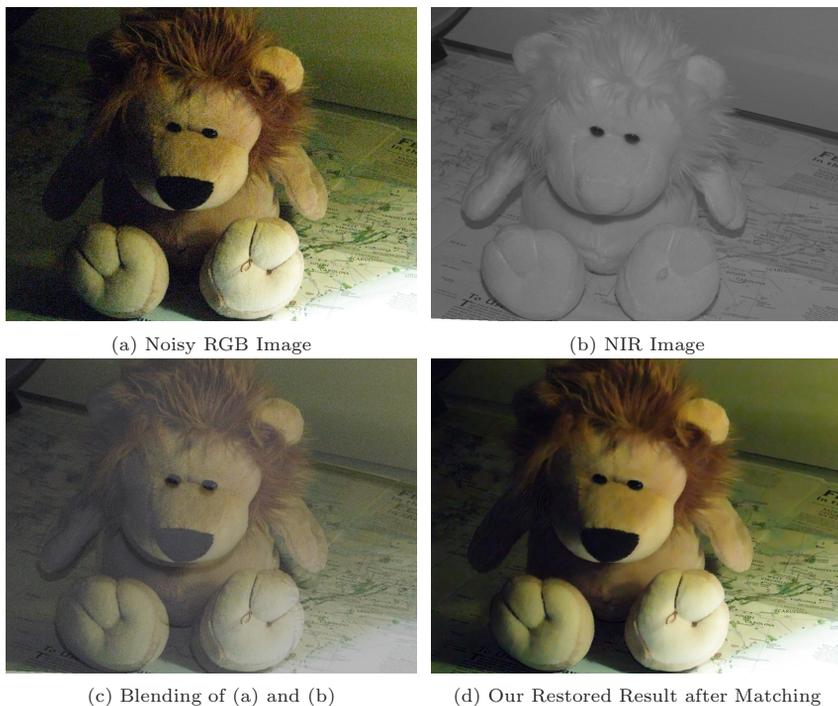
(f) Blending Result of (a) and (d)

(g) Blending Result of (a) and (e)

(h) Our Restoration Result of (a)

Fig. 11. Multispectral image restoration example. (a) and (b) are the unmatched noisy RGB and clean NIR images respectively. (d) and (e) are the results of SIFT Flow and our method respectively. (h) is the restoration result by the method of [8] using our matching result. (c), (f) and (g) are the blending results to illustrate if any misalignment exists.

(a) Noisy RGB Image

(b) NIR Image

(c) Blending of (a) and (b)

(d) Our Restored Result after Matching

Fig. 12. Multispectral image restoration example. (a) and (b) are the unmatched noisy RGB and clean NIR images respectively. (c) is the blending of (a) and (b). (d) is the restoration result by the method of [8] using our matching result.

(a) NIR Image   (b) RGB Image   (c) SIFT Flow Result   (d) Our Result
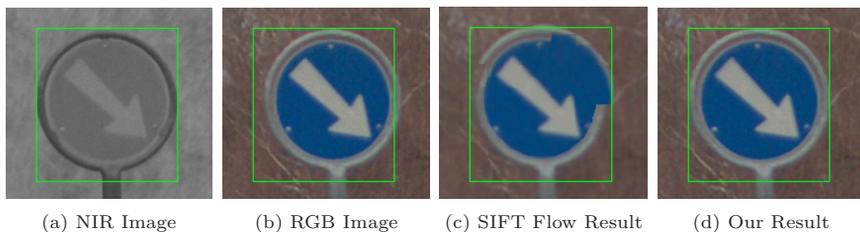
Fig. 13. RGB image and NIR image matching example. (a) and (b) are the unmatched NIR and RGB images respectively. (c) is the warping result of (b) according to (a) by SIFT Flow [2] while (d) is our result.