Appendix A: Solver

We reproduce our optimization problem below for reference:

$$\begin{aligned}\min_{\mathbf{A},\mathbf{B}} \; & \left\|\mathbf{I}^{\text{cfa}} - \mathbf{D}\left(\mathbf{A}\odot\mathbf{I}^{\text{ref}} + \mathbf{B}\odot(\mathbf{1}-\mathbf{I}^{\text{ref}})\right)\right\|_2^2 \\ & +\lambda_1\|\mathbf{W}\odot\nabla\mathbf{A}\|_1 + \lambda_2\|\mathbf{W}\odot\nabla\mathbf{B}\|_1 \\ & +\gamma_1\sum_{l\neq k}\|\mathbf{W}\odot(\nabla\mathbf{A}_l\odot\mathbf{A}_k - \nabla\mathbf{A}_k\odot\mathbf{A}_l)\|_1 \\ & +\gamma_2\sum_{l\neq k}\|\mathbf{W}\odot(\nabla\mathbf{B}_l - \nabla\mathbf{B}_k)\|_1,\end{aligned} \tag{1}$$

where $\mathbf{A},\mathbf{B},\mathbf{I}^{\text{ref}} \in \mathbf{R}^{3n}$ and $\mathbf{I}^{\text{cfa}} \in \mathbf{R}^{n}$ are in the vectorized form with n being the total number of pixels. $\mathbf{D} \in \mathbf{R}^{n\times 3n}$ denotes the Bayer downsampling matrix. ∇ denotes the gradient operator for the underlying images and $\mathbf{W}$ is a per-pixel weight matrix that matches the output dimension of ∇.

Writing $\mathbf{x} = \begin{bmatrix}\mathbf{A}; & \mathbf{B}\end{bmatrix} \in \mathbf{R}^{6n}$ and expressing the element-wise multiplication as matrix multiplication with a diagonal matrix, the above problem can be reformulated as

$$\min_{\mathbf{x}} \; f(\mathbf{x}) + g(\mathbf{K}\mathbf{x}), \tag{2}$$

with $f(\mathbf{x}) = \left\|\mathbf{I}^{\text{cfa}} - \mathbf{D}\begin{bmatrix}\text{diag}_{\mathbf{I}^{\text{ref}}} & \text{diag}_{\mathbf{1}-\mathbf{I}^{\text{ref}}}\end{bmatrix}\mathbf{x}\right\|_2^2$ and $g(\cdot) = \|\cdot\|_1$. Here $\text{diag}_{\mathbf{v}}$ represents a diagonal matrix with diagonal entries given by a vector $\mathbf{v}$. The combined matrix $\mathbf{K}$ is obtained by stacking the component matrices. More specifically,

$$\mathbf{K} = \begin{bmatrix}\lambda_1\text{diag}_{\mathbf{W}}\nabla & \mathbf{0} \\ \mathbf{0} & \lambda_2\text{diag}_{\mathbf{W}}\nabla \\ \gamma_1\mathbf{K}_{\mathbf{A}} & \mathbf{0} \\ \mathbf{0} & \gamma_2\mathbf{K}_{\mathbf{B}}\end{bmatrix}, \tag{3}$$

where $\mathbf{K}_{\mathbf{A}}$ and $\mathbf{K}_{\mathbf{B}}$ compute the difference between (scaled) gradients of different channels of $\mathbf{A},\mathbf{B}$ respectively.

The minimization problem (2) has a standard form for which a number of non-linear solvers are available. In our implementation we have used the linearized ADMM algorithm with steps included below for completeness.

Algorithm 1 Linearized ADMM

Input: $\mathbf{I}^{\text{ref}},\mathbf{I}^{\text{cfa}}$; Parameters μ,α satisfying $0 < \mu \leq \alpha/\|\mathbf{K}\|_2$.

Repeat until convergence:

1: $\mathbf{x}^{k+1} = \underset{\mathbf{x}}{\text{argmin}}\; f(\mathbf{x}) + \frac{1}{2\mu}\left\|\mathbf{x} - \left(\mathbf{x}^k - \frac{\mu}{\alpha}\mathbf{K}^T(\mathbf{K}\mathbf{x}^k - \mathbf{z}^k + \mathbf{u}^k)\right)\right\|_2^2$,

2: $\mathbf{z}^{k+1} = \underset{\mathbf{z}}{\text{argmin}}\; g(\mathbf{z}) + \frac{1}{2\alpha}\left\|\mathbf{z} - (\mathbf{K}\mathbf{x}^{k+1} + \mathbf{u}^k)\right\|_2^2$,

3: $\mathbf{u}^{k+1} = \mathbf{u}^k + \mathbf{K}\mathbf{x}^{k+1} - \mathbf{z}^{k+1}$.

Output: $\mathbf{x}^* = \begin{bmatrix}\mathbf{A}^*; & \mathbf{B}^*\end{bmatrix}$ and $\mathbf{I}^{\text{out}} = \mathbf{A}^*\odot\mathbf{I}^{\text{ref}} + \mathbf{B}^*\odot(\mathbf{1}-\mathbf{I}^{\text{ref}})$.

Appendix B: Implementation Details of W-Matrix Reconstruction

The flow consistency is measured for each pixel by checking if its L2 distance between forward and backward flows is below 2 pixels.

The visual similarity is evaluated between the reference image and the green channel of color images used in alignment. Texture similarity can be identified by comparing SIFT descriptor [3] on gradient images. If the L1-norm SIFT difference of two aligned pixels is less than $\lambda_{sift} = 2$, the two pixels are identified to be visually similar and thus reliably aligned. Before computing the SIFT descriptor, the image gradients are smoothed by guided filter [2] with a window size of 4 pixels and 0.03 degree of smoothing. Two textureless regions can also be perceived as being similar to each other, if both of their local gradients are smaller than a threshold of 0.06.

For detecting occluded pixels, the difference between image edge maps is evaluated with structural similarity (SSIM) [5] in which only the structure comparison module is used. The two aligned pixels are deemed unreliable if the SSIM is smaller than 0.3. The image edge is detected via SDE detector [1], which is also used in EpicFlow [4]. But edge-based detection cannot deal with relatively large occluded regions. Additionally, flow gradients are used to detect large occluded regions. In a two-camera stereo system, where the monochrome camera is fixed to always be located either on the left or on the right of the RGB camera, occlusion artifacts always appear on the same side of an object boundary. This means that, for a chosen camera, we only need to deal with either positive or negative flow gradients. Therefore we first discard flow gradients with the unwanted sign depending on the camera position. Absolute values of gradients are used for subsequent processing. The absolute gradients smaller than 0.5 is set as 0. We remove small non-zero region with a threshold of 100 pixels. This can remove flow gradients that result from miscalculated and noisy flow fields at, for example, textureless regions instead of at real depth discontinuities. Then a 1-D Gaussian filter with standard deviation of $\sigma = 40$ along x-axis is used to dilate the processed absolute gradient field. A pixel is occluded, if its dilated gradient exceed 0.1 and the L2 distance between forward and backward flows exceeds 15 pixels. For highly regular-textured regions, if L1 SIFT difference is smaller than $\Lambda_{sift} = 5$, which is loosened from $\lambda_{sift} = 2$ and absolute flow gradient is smaller than 0.03, the two pixels are well-aligned.

In this end, the binary mask of reliable pixels are filtered using median filter with 15×15 windows to remove thresholding noise. In **W**-matrix, the excluded pixels are given low weight, e.g. 0.05, to ensure the reconstructed pixels are similar to the upsampled raw pixels rather than the reference artifact pixels, while encouraging reliable pixels to propagate **A** and **B** to these unreliable pixels. The reliable pixels are set to have weight 1. **W** is finally filtered by Gaussian filter with a standard deviation of 2 pixels to smoothen the boundaries.

Appendix C: Robustness of W-Matrix Construction

Our algorithm is robust to underlying artifacts in the reference image. The **W**-matrix is robustly and conservatively constructed through a relatively complete heuristic rules. We run our W-matrix construction algorithm with randomly and independently perturbed parameters on Middlebury dataset 2014. Once the image pair is aligned, we calculate PSNR on reliably and unreliably aligned pixels identified by the W-matrix, which is shown in Fig. 1

(Left). We found that perturbing each parameter by an average of 20 % (maximum 40 %) has insignificant impact on reference images.

Even if some artifact pixels are wrongly weighted too high in the regularization term by $\mathbf{W}$, our algorithm still produces high-quality reconstruction results using a single set of parameters since our image formation model has considered underlying artifacts in the reference images. We do not assume the reference monochrome image is perfectly aligned to the color image. We run the perturbation experiments on datasets of our captured image pairs. Fig. 1 (Right) shows the parameter perturbation has even less impact on the reconstructed images quality. The maximum and minimum PSNR of the reconstructed images are almost unchanged.

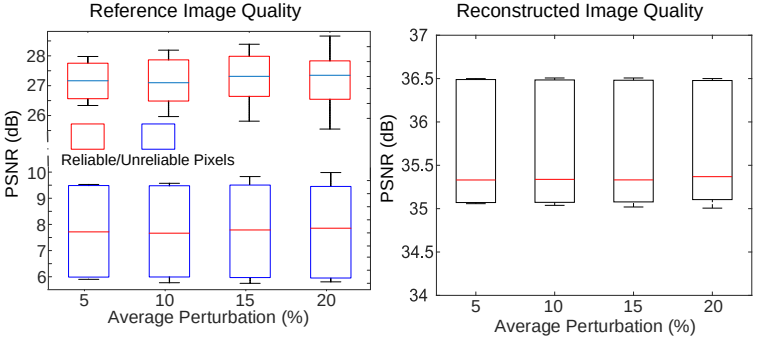

Figure 1: Box plot of PSNR with respect of parameter perturbation. Middle lines in the box indicate the mean, the upper and lower lines indicate the maximum and minimum. The length of the box suggests PSNR variance.

References

[1] Piotr Dollár and C Lawrence Zitnick. Structured forests for fast edge detection. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1841–1848. IEEE, 2013.

[2] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence*, 35(6):1397–1409, 2013.

[3] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[4] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. In *Computer Vision and Pattern Recognition*, 2015.

[5] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.