Supplementary Materials for KeystoneDepth: History in 3D

Xuan Luo[1] Yanmeng Kong[1] Jason Lawrence[2] Ricardo Martin-Brualla[2] Steven M. Seitz[1,2]
[1]University of Washington [2]Google Research
{xuanluo, yk57, seitz}@cs.washington.edu, {jdlaw, rmbrualla}@google.com

1. Single-Plate Stereo Rectification

In this section we provide additional analyses and implementation details for our single-plate stereo rectification method. Sec. 1.1 describes how we extract correspondences in the absence of ground truth for all of our experiments. Sec. 1.2 demonstrates the rectification quality of the *KeystoneDepth* collection by analyzing the amount of vertical parallax between the stereo pairs. And finally, Sec. 1.3 gives more details on how the affine model is a degenerate case.

1.1. Correspondence Extraction

To find reliable image correspondences between the left and right sides of a stereograph, we use a combination of SIFT [3] and optical flow [1]. We observe that SIFT features have the advantage of avoiding ambiguous matches in textureless regions, while FlowNet2 avoids outliers by considering non-local image context. Based on these observations, we first compute optical flow and filter matches with forward-backward consistency threshold of 1 pixel. We then extract SIFT matches and keep a SIFT match only if it passes a ratio test threshold of 0.7 [4] and its corresponding point is less than 3 pixels away from its flow match.

1.2. Rectification Quality Assessment

We assessed the performance of our rectification algorithm by analyzing the amount of residual vertical parallax within the rectified artifact-free stereo pairs, across the entire *KeystoneDepth* collection. Perfect image rectification should result in stereo pairs with zero vertical parallax. The amount of vertical parallax over all matches across the entire collection has a mean of 0.26 pixels and a standard deviation of 0.33 pixels, and more than 96.1% percent of matches have a vertical parallax below 1 pixel.

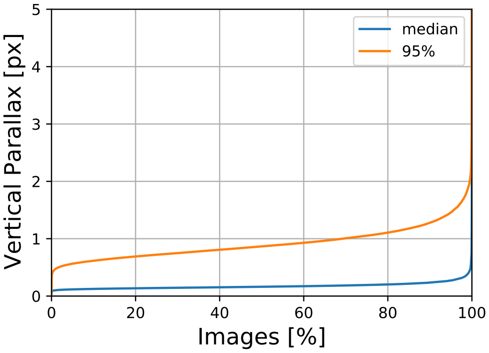

Figure 1: **Distribution of per-image median and 95-percentile of vertical parallax in *KeystoneDepth* collection**. 99.95% of images have a median vertical parallax below 1 pixel. 68.4% and 99.4% of images have a per-image 95-percentile of vertical parallax of below 1 pixel and 2 pixels, respectively. Note that our automated method for computing the image matches used to calculate these statistics is not perfect, and so some amount of matching errors and outliers inflate these values.

Fig. 1 shows how these errors are distributed across images. It plots the percentage of images whose median (blue curve) and 95th-percentile matches (orange curve) are below a specific amount of vertical parallax. It shows that 99.95% of images have a median vertical parallax below 1 pixel; and that 68.4% and 99.4% of images have a per-image 95-percentile of vertical parallax of below 1 pixel and 2 pixels, respectively. Note that our automated method for computing the image matches (Section 1.1) used to calculate these statistics is not perfect, and so some amount of matching errors and outliers inflate these values.

The main take-away from all of these statistics is that vertical parallax is not a significant issue in the vast majority of rectified images, and the dataset is therefore well-suited for stereo algorithms. Nevertheless, we found that an optical flow method (FlowNet2) outperformed most state of the art stereo methods for disparity estimation. While the optical flow method does do better in the few cases where vertical parallax is present, these failures still occur even when the vertical disparity is sub-pixel, and thus we attribute FlowNet2's better performance to its invariance to noise and exposure characteristics of antique images. We expect the *KeystoneDepth* dataset will encourage development of stereo algorithms that are robust to these specific types of image artifacts.

1.3. Degenerate Case

As mentioned in the main paper, one degenerate case is when the two images are approximately the same up to a 2D affine transformation. More specifically, if the following affine transformation holds for all corresponding points $\mathbf{u_l}, \mathbf{u_r}$ in the left and right images respectively,

$$u_l = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} u_r, \tag{1}$$

then the the stereo images can be rectified by $\forall\gamma$,

$$H_l = \begin{bmatrix} 1 & -\gamma & 0 \\ \gamma & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, H_r = \begin{bmatrix} e & -d & 0 \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} + \gamma \begin{bmatrix} b & -a & 0 \\ a & b & c \\ 0 & 0 & 1 \end{bmatrix}.$$

2. Implementation Details

In this section, we provide implementation details for training the inpainting networks (Sec. 2.1) and generating boundary maps (Sec. 2.2).

2.1. Training

We trained our intensity and depth inpainting networks using the *KeystoneDepth* collection with a train, validation and test split ratio of $94\%, 5\%$ and 1%, using the Adam solver [2] with $\beta_1 = 0.9$, $\beta_2 = 0.999$. We use a batch size of 4, a learning rate of 2×10^{-4}, and train for 53 epochs. During training, all images have a spatial resolution of 512×512. Training required about two days with a Nvidia GTX 1080Ti GPU.

2.2. Boundary Map

Recall that we introduce a ”boundary map” as an additional input to our inpainting networks in order to encourage them to produce sharp transitions between foreground and background scene elements. This is a binary mask where pixels having a value of 1 correspond to the ”foreground side” of boundaries between the foreground and background, and 0 elsewhere. Defining the mask in this way has the desirable effect of causing the boundary map to remain intact and consistent when it is projected to another camera viewpoint. Specifically, we apply a Laplacian filter to the disparity map and define the boundary mask to be on the positive side. A second criterion is that the relative difference in disparities between nearby pixels must exceed a threshold. Given a disparity map D, we set the boundary map at pixel (u, v) to be one if and only if it passes the following two tests, and zero otherwise:

1. $D(u, v) - GaussBlur(D, (k_s, k_s))(u, v) > \epsilon$, with a spatial kernel size of $k_s = 15$, and threshold $\epsilon = 10^{-9}$.
2. Since zero disparity is prone to numeric error, we compute $\hat{D} = max(3, D)$. Then we threshold the relative difference by testing whether $\frac{\hat{D}(u,v) - erode(\hat{D},(k_e,k_e))(u,v)}{\hat{D}(u,v)} > \delta$, where we apply a morphological erosion operator on the disparity map with kernel $k_e = 5$ and apply a threshold of $\delta = 0.1$.

References

[1] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE conference on computer vision and pattern recognition (CVPR)*, volume 2, page 6, 2017. 4321

[2] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 4322

[3] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999. 4321

[4] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 4321