Depth estimation with a practical camera

Arnav V. Bhavsar
arnav.bhavsar@gmail.com

A. N. Rajagopalan
http://www.ee.iitm.ac.in/~raju

Image Processing and Computer Vision Lab
Indian Institute of Technology, Madras
Chennai, India

We propose a framework for depth estimation from a set of calibrated images, captured with a moving camera with varying parameters. Our framework respects the physical limits of the camera, and considers various effects such as motion parallax, defocus blur, zooming and occlusions which are often unavoidable. In fact, the stereo [1] and the depth from defocus [2] are essentially special cases in our more general framework. The relationships between blur-depth and motion-depth strongly couple the two effects of blurring and motion parallax for depth estimation.

The observed images g_is can be modeled to be warped and blurred manifestations of an ideally non-blurred image f as

$$g_i(n_1,n_2) = \sum_{l_1,l_2} h_i(n_1,n_2,\sigma_i,\theta_{1i}(l_1),\theta_{2i}(l_2)) \cdot f(\theta_{1i}(l_1),\theta_{2i}(l_2)) + \eta_i(n_1,n_2) \qquad (1)$$

where $\theta_{1i}(l_1)$ and $\theta_{2i}(l_2)$ denote the geometric transformation, and the kernel $h_i(n_1,n_2,\sigma_i,\theta_{1i}(l_1),\theta_{2i}(l_2))$ blurs a pixel at $(\theta_{1i}(l_1),\theta_{2i}(l_2))$ in the i^{th} image $f(\theta_{1i}(l_1),\theta_{2i}(l_2))$.

For simplicity, in this work we consider the camera motion to be purely translation. Hence, the geometric transformation between the reference and the i^{th} view also considering variation in the lens to image plane distance can be expressed as,

$$\theta_{1i}(l_1) = \frac{v_r l_1 Z + v_i t_{xi}}{Z + t_{zi}} \quad \text{and} \quad \theta_{1i}(l_2) = \frac{v_r l_2 Z + v_i t_{yi}}{Z + t_{zi}} \qquad (2)$$

where t_{xi}, t_{yi} and t_{zi} are the camera translations, v_1 and v_i are the lens to image plane distances in the reference and the i^{th} view, respectively, $v_r = v_i/v_1$ and Z is the depth of the 3D point in the reference coordinates that projects at the pixel (l_1,l_2).

The blur kernel can be modeled by a 2D Gaussian function as

$$h_i(n_1,n_2,\sigma_i,\theta_{1i}(l_1),\theta_{2i}(l_2)) = \frac{1}{2\pi\sigma_i^2}\exp\left(-\frac{(n_1-\theta_{1i}(l_1))^2+(n_2-\theta_{2i}(l_2))^2}{2\sigma_i^2}\right) \qquad (3)$$

The blur parameter σ is related to the absolute depth from the lens. As the camera translates along the optical axis by t_{zi} for the i^{th} image, the depth of a 3D point in the i^{th} view is $Z \pm t_{zi}$. The blur parameter σ_i, is related to this depth through the aperture r_i, focal length f and v_i.

$$\sigma_i = \rho r_i v_i \left(\frac{1}{f} - \frac{1}{v_i} - \frac{1}{Z \pm t_{zi}}\right) \qquad (4)$$

Note that the blur kernel is centered at $(\theta_{1i}(l_1),\theta_{2i}(l_2))$. Thus, according to the above model the blur kernel is formed around the point where the central ray projects on to the image plane. Hence, the position of the blur kernel is also warped in the i^{th} image.

We formulate the estimation in a MAP framework and use belief propagation (BP) to compute a solution [3]. The BP algorithm treats the image as a graph, with the nodes as pixels. It works by passing messages between neighbouring nodes and computing beliefs to decide about the label that a node takes. The messages and the beliefs are functions of data cost that depends on the observation and the prior cost between the nodes.

To define the data cost we relate the reference and i^{th} image as

$$g_1(n_1,n_2) = h_{ri}(\sigma_i,n_1,n_2) * g_i(n_1,n_2) \qquad (5)$$

Here, h_{ri} signifies the relative blur kernel which corresponds to blur parameter $\sqrt{\sigma_1^2-\sigma_i^2}$ which can be related to Z using (4). The symbol $*$ denotes convolution. The data cost is then defined as

$$D(n_1,n_2) = |g_1(n_1,n_2) - h_{ri}(\sigma_i,n_1,n_2) * g_i(n_1,n_2)| \qquad (6)$$

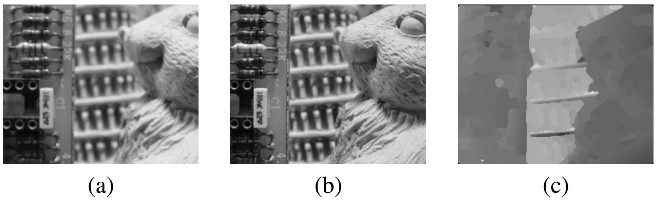

Figure 1: Real experiment. (a) is horizontally translated with respect to (b) and is also captured with a larger aperture than that used for (b). Note that in the estimated depth map (c), the depth discontinuities are well localized and fine depth variations are also captured (e.g. on the Pisa tower-model and the circuit board)

The prior cost constrains the neighbouring nodes to have similar labels. We define the smoothness prior as a truncated absolute function

$$V_p(n_1,n_2,m_1,m_2) = \min(|Z(n_1,n_2) - Z(m_1,m_2)|, T) \qquad (7)$$

where, (n_1,n_2) and (m_1,m_2) are neighbouring nodes in a 4-connected neighbourhood. The truncation allows discontinuities in the solution.

We incorporate the notion of visibility in the above data term using a binary visibility function V which modulates the datacost as

$$D(n_1,n_2) = V_i(n_1,n_2) \cdot |g_1(n_1,n_2) - h_{ri}(\sigma_i,n_1,n_2) * g_i(n_1,n_2)| \qquad (8)$$

In the beginning, all pixels are considered visible. After the first iteration, the visibility V_i for the i^{th} view is computed by warping the current estimate of the depth in that view.

We also exploit the visual cue from color image segmentation that has been quite successful in the stereo vision domain [4]. Before beginning the BP iterations, we compute the color segmented image and a reliability map that specifies at which pixels the depth estimates are more ‘reliable’. The first BP iteration is run without using the segmentation cue. We then compute a (partially) plane-fitted depth map that uses the current depth estimate, the segmented image and the reliability map. This plane-fitted depth map is used to regularize the data term in subsequent iterations. The modified data cost with this regularization is expressed as

$$D(n_1,n_2) = V(n_1,n_2) \cdot |g_1(n_1,n_2) - h_{ri}(\sigma_i,n_1,n_2) * g_i(n_1,n_2)| + w \cdot |Z(n_1,n_2) - Z_p(n_1,n_2)| \qquad (9)$$

where Z_p denotes the plane-fitted depth map and the binary weight w is 0 if the pixel is reliable and 1 if it is not. The regularization term constrains the estimated values at unreliable pixels to remain close to those in the plane-fitted depth map.

Thus our BP algorithm that considers both defocus and motion effects, along with constraints from the MRF prior, visibility and segmentation cue yields a solution with good discontinuity localization and detail preservation, and can also handle low-textured regions.

[1] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7–42, 2002.

[2] A. N. Rajagopalan and S. Chaudhuri. *Depth from defocus: A real aperture imaging approach*. Springer-Verlag New York, Inc., New York, 1999.

[3] P. Felzenszwalb, D. Huttenlocher. Efficient belief propagation for early vision. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1996)*, 1:261–268, 2004.

[4] Q. Yang, L. Wang, R, Yang, H. Stewenius and D. Nister. Stereo matching with color-weighted correlation, heiarchical belief propagation, and occlusion handling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(3):492–504, 2009.