A first-order optimization method for learning to reconstruct opacity in computational imaging

Keith Dillon

May 9, 2021

Abstract

Unknown self-occlusion in a scene with opaque objects causes the multiview reconstruction problem to become ill-posed and nonlinear. In this paper we describe a nonlinear optimization method for simultaneously reconstructing the object and occlusion. The approach uses a simple learning stage which forms hypotheses for scene opacity given locally-optimal reconstruction estimates. This is combined with a first-order projected-gradient method for imposing physical consistency. Results are demonstrated for simulated examples.

Introduction

Multiview reconstruction is a form of computational imaging where a diverse set of views of the same object are used to reconstruct a three-dimensional image of the object [1]. Assuming the view directions are known and the scene is fixed, it is a relatively straightforward inverse problem as the forward model can be ascertained. If there are opaque objects occluding other parts of a scene, however, this inverse problem can become ill-posed and nonlinear. Consider a simple scene consisting of one object occluding another; the resulting sensor data is clearly not the sum of signals produced by the objects individually, but only of the object in front.

In problems known as computational photography, multi-view reconstruction, or volumetric reconstruction [1], the goal is to use multiple two-dimension images (e.g. photographs) from different views, to reconstruct a three-dimensional object. This might be desired directly for the formation of computer models [2], or for applications such as computer vision and compression for 3D video [3]. Perhaps the most common approaches historically were heuristic techniques such as voxel coloring [4].

Modern approaches to reconstruct such computational imaging reconstruction problems can be viewed as forms of nonlinear optimization [5]. Within this framework, constraints can be employed to enforce the physical properties of optics with opacity [6], and regularization can be employed to address the ill-posedness [7].

	Air	Solid	Range
Attenuation	$a(\mathbf{r}) = 1$	$a(\mathbf{r}) = 0$	$1 \geq a(\mathbf{r}) \geq 0$
Brightness	$s(\mathbf{r}) = 0$	$s(\mathbf{r}) > 0$	$s(\mathbf{r}) \geq 0$

Table 1: Unknowns for points in empty space ("air"), in opaque solid, and the valid range.

The problem remains challenging, however, as the amount of data is often very limited and regularized imposes fairly crude presumptions, often leading to poor results. Recently, researchers have applied machine learning methods to improve on this process. In the most direct approach, the inverse model itself might be estimated from data via supervised learning [8], which may be rather extreme as it discards usable knowledge of the forward model. At an opposite extreme, data might be used solely to model a better prior for images, producing a more specialized form of regularization. Here, we take a complementary approach where we train a model to impose the constraints which are the basis for opacity in the optimization, and a significantly more problematic aspect of the problem due to their non-convexity.

Method

We use the ray-optics model of [9], which described a scene containing unknown point sources and attenuation. If we define the vector $\mathbf{a}$ as a sufficiently-sampled version of the attenuation at each point $a(\mathbf{r})$, and $\mathbf{s}$ similarly from the source distribution $s(\mathbf{r})$, then our unknown is the vector $\mathbf{x}$ formed by concatenating $\mathbf{a}$ and $\mathbf{s}$.

$$\mathbf{x} = \begin{pmatrix} \mathbf{a} \\ \mathbf{s} \end{pmatrix} = \begin{pmatrix} a(\mathbf{r}_1) \\ \vdots \\ a(\mathbf{r}_K) \\ s(\mathbf{r}_1) \\ \vdots \\ s(\mathbf{r}_K) \end{pmatrix} \tag{1}$$

K is the total number of pixels (or voxels) in the sampled image of the object. The forward model that relates the detected light samples $\mathbf{b}$ to the unknown object $\mathbf{x}$ is given by

$$\mathbf{b} = \mathbf{C}\ \exp\Big(\mathbf{E} \log \mathbf{x}\Big). \tag{2}$$

The matrices $\mathbf{C}$ and $\mathbf{E}$ describe the (known) ray propagation between points in the scene and points on the detector. The log and exponential are taken element-wise over vectors.

In this model, the unknowns are only valid for certain ranges (e.g., negative brightness is not allowed in intensity imaging). The ranges and values relevant to a scene containing opaque objects is given in Table 1. Note that for a scene containing only opaque objects and empty space, the attenuation would be constrained to take only integer values (zero or one), making for a non-convex constraint. In [9], this constraint was relaxed to convex inequality constraints (the range column in Table 1).

Generally, we form the optimization problem of Eq. (3), which regularizes points to values describing air.

$$\min_{\mathbf{x}} \{\|\mathbf{b} - \mathbf{C}\exp\{\mathbf{E}\log\,\mathbf{x}\}\|_2^2 + \mu_1\|\mathbf{x} - \mathbf{x}_{air}\|_1 + \mu_2\|\mathbf{x} - \mathbf{x}_{air}\|_2^2\} \quad \mathbf{x} \in F \tag{3}$$

The scalar constants μ_1 and μ_2 are regularization parameters. F denotes the feasible set, depending on how we impose the requirements of Table 1. $\mathbf{x}_{air}$ is given by

$$\mathbf{x}_{air} = \begin{pmatrix} \mathbf{1}_K \\ \mathbf{0}_K \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{4}$$

To solve this problem, we build on the following first-order algorithm. The range constraint in step 5.5.2 is imposed based on Table 1 with a small nonzero ϵ in place of zero, to avoid issues with the logarithm. Adaptive restarts [10] are used in step 5.5 to reset the acceleration.

1. Choose starting point $\mathbf{x}^{(0)}$, stepsize t, convergence criterion r.
2. Set $\mathbf{y}^{(0)} = \mathbf{x}^{(0)}$.
3. Set $\mathbf{g}^{(0)} = \nabla f(\mathbf{y}^{(0)})$.
4. Set $k = 1$.
5. **while** *not converged* **do**
   - 5.1. Set $\mathbf{x}^{(k)} = \mathbf{y}^{(k-1)} - t\mathbf{g}^{(k-1)}$
   - 5.2. Clamp $\mathbf{x}$ to relaxed constraints $\epsilon \leq \mathbf{x} \leq x_{max}$
   - 5.3. Set $\mathbf{y}^{(k)} = \mathbf{x}^{(k)} + \theta \cdot (\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)})$
   - 5.4. Set $\theta = \frac{2}{1+\sqrt{1+4/\theta^2}}$
   - 5.5. **if** $(\mathbf{y}^{(k)} - \mathbf{x}^{(k)})^T(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}) > 0$ **then**
     - 5.5.1. Set $\theta = 0$
     - 5.5.2. set $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)}$

     end
   - 5.6. Set $\mathbf{g}^{(k)} = \nabla f(\mathbf{y}^{(k)})$
   - 5.7. Set $k = k + 1$

end

Algorithm 1: Accelerated projected-gradient algorithm with restarts.

The rationale for restarts is that the acceleration appears to have overshot the optimum, so the accumulated direction information is cleared. This can be viewed as an outer loop where, at each inner iteration, a rough estimate of $\mathbf{x}$ is formed using relatively local information. We extend this approach to impose non-convex constraints at the resets. Each time the acceleration is reset, we also impose the integer constraint on the attenuation (forcing values to be either zero or one), and the consistently requirements between the attenuation and brightness.

We use a network consisting of a single convolutional layer to perform the constraint enforcement. The values for attenuation and brightness at each point are estimated using the pixel

values in the local vicinity. Parameters are determined in an offline stage using supervised learning; the true values for opacity and air in an image are the target, while the approximate values reached by Algorithm 1 at the restarts are used as the training data. Since it is a single-layer network, binary activation functions can be used with the perceptron learning algorithm [11] to estimate the weights $\mathbf{w}$ and the bias b for each neuron.

Results

We tested the approach with a two-dimensional simulation consisting of a number of opaque objects, viewed from a range of directions between zero and 359 degrees, at one-degree increments (so 360 total views). The true brightness images for several of the randomly-generated test cases are given in Fig. 1. The opacity for each case is simply a binarized

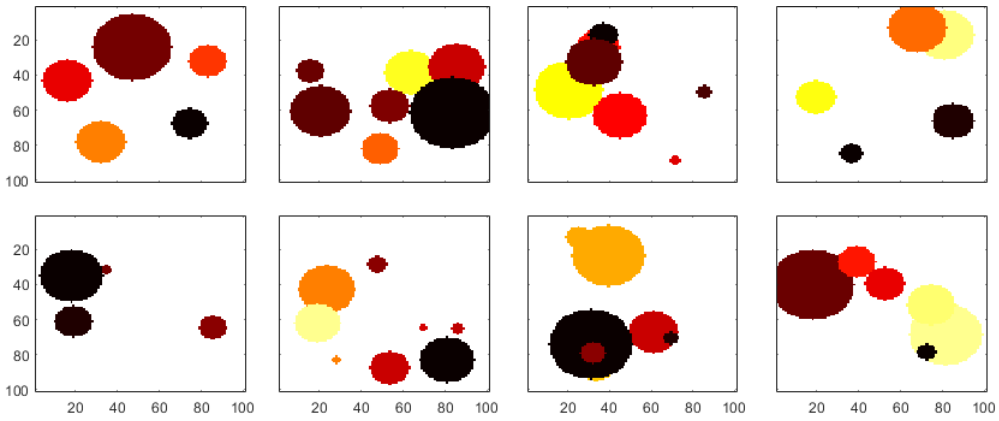

Figure 1: Brightness values for test cases.

version of the same image.

First we demonstrate the learning stage is functioning properly by training an overfit case. We use the largest possible convolution kernel (all pixels in the image) and train on a single image, then test performance on that same image versus a different image. This result is shown in Fig. 2, where we simply used an inverted version of the same image to get the failed test case.

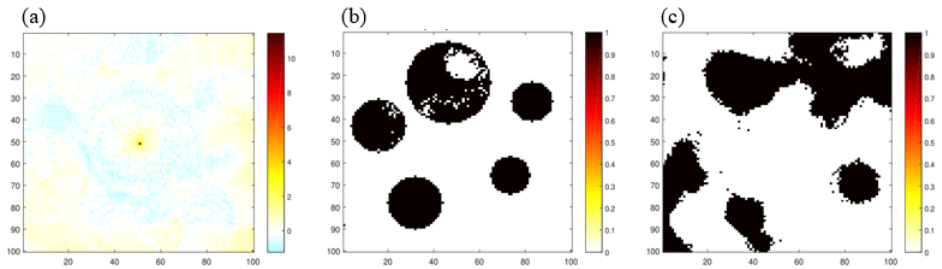

Figure 2: Overfitting test: (a) convolution kernel, (b) reconstructed image using same image as training, (c) reconstructed image using inverted (flipped vertically) training image.

Next we imposed rotational symmetry on the convolution kernel, given that we expect the system to be invariant to rotations of an object (just as the convolutional structure imposes translation invariance). We also reduced the size of the kernel. The result is given in Fig. 3.

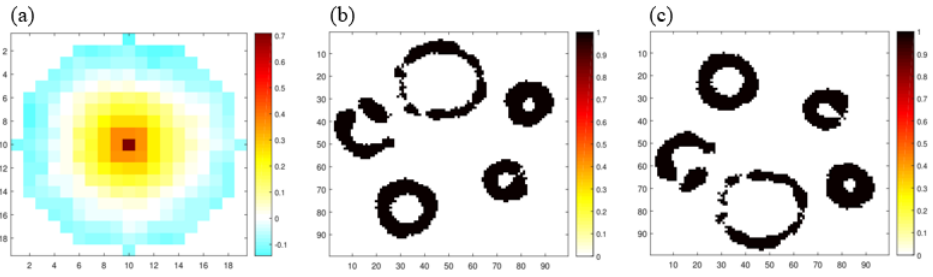

Figure 3: Overfitting test using rotational symmetry constraint and smaller convolution kernel: (a) convolution kernel, (b) reconstructed opacity image using same image as training, (c) reconstructed opacity image using inverted (flipped vertically) training image.

Finally, we trained this system using the entire set of test cases, with the first case held out as a test set. We also tested different-sized convolution kernels. The result is given in Fig. 4, with a 5×5 convolution kernel, compared to the same reconstruction algorithm performed without the pixel learning stage (i.e., Algorithm 1), where we see a noticeable improvement in the estimate. The size of the convolution kernel did not have much impact; all were

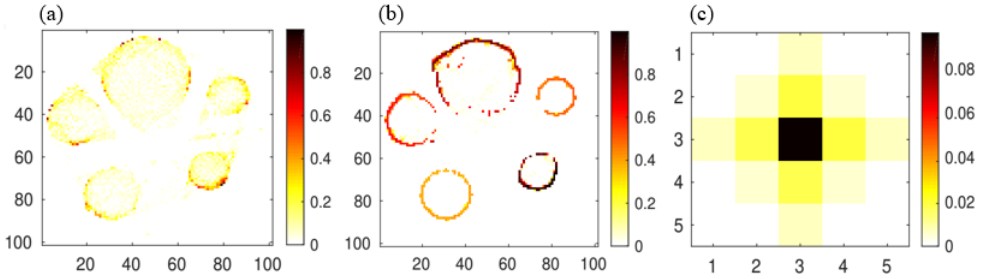

Figure 4: Full training set test: (a) brightness images reconstructed without pixel learning, (b) brightness image reconstructed with pixel learning, (c) convolution kernel.

essentially of the same shape. Note that the estimated images contain hollow shapes, as the interior are occluded and the regularization drives unseen pixels to transparency (i.e., "air" points).

Discussion

We achieved a noticeable improvement in simulated reconstruction results by incorporating a simple learning network into a first-order optimization algorithm. The convolution kernel optimized through supervised learning resembled a basic smoothing kernel, though if larger sizes were used then the result appeared to incorporate an edge-detection component. Deeper

networks could potentially provide additional benefit, as the common semantic properties of the objects could be exploited. The method could also be integrated with techniques for optimizing the regularization.

References

[1] C. R. Dyer, "Volumetric Scene Reconstruction from Multiple Views," in *Foundations of Image Understanding*, L. S. Davis, Ed., *The Kluwer International Series in Engineering and Computer Science* **628**, 469–489, Springer US (2001).

[2] M. Pollefeys and L. V. Gool, "From images to 3D models," *Commun. ACM* **45**, 50–55 (2002).

[3] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, "Multiview Imaging and 3DTV," *Signal Processing Magazine, IEEE* **24**, 10 –21 (2007).

[4] S. M. Seitz and C. R. Dyer, "Photorealistic Scene Reconstruction by Voxel Coloring," *International Journal of Computer Vision* **35**(2), 151–173 (1999).

[5] K. Dillon, *Optimization in Computational Imaging and Inverse Problems*. PhD thesis, UNIVERSITY OF CALIFORNIA, SAN DIEGO (2014).

[6] K. Dillon and Y. Fainman, "Computational confocal tomography for simultaneous reconstruction of objects, occlusions, and aberrations," *Applied Optics* **49**, 2529–2538 (2010).

[7] K. Dillon, Y. Fainman, and Y.-P. Wang, "Computational estimation of resolution in reconstruction techniques utilizing sparsity, total variation, and nonnegativity," *Journal of Electronic Imaging* **25**(5), 053016–053016 (2016).

[8] G. Barbastathis, A. Ozcan, and G. Situ, "On the use of deep learning for computational imaging," *Optica* **6**, 921–943 (2019). Publisher: Optical Society of America.

[9] K. J. Dillon and Y. Fainman, "Computational Lightcurve Imaging," in *Computational Optical Sensing and Imaging*, *OSA Technical Digest (online)*, CTu4B.3, Optical Society of America (2012).

[10] B. O'Donoghue and E. Candès, "Adaptive Restart for Accelerated Gradient Schemes," *Foundations of Computational Mathematics* **15**, 715–732 (2015).

[11] S. Gallant, "Perceptron-based learning algorithms," *IEEE Transactions on Neural Networks* **1**, 179–191 (1990). Conference Name: IEEE Transactions on Neural Networks.