

# Shape from Caustics: Reconstruction of 3D-Printed Glass from Simulated Caustic Images: Supplementary Material

Marc Kassubeck<sup>1</sup> Florian Bürger<sup>2</sup> Susana Castillo<sup>1</sup> Sebastian Stiller<sup>2</sup> Marcus Magnor<sup>1</sup>

<sup>1</sup> Institut für Computergraphik, TU Braunschweig [graphics.tu-bs.de](mailto:graphics.tu-bs.de)

<sup>2</sup> Institute for Mathematical Optimization, TU Braunschweig [mo.tu-bs.de](mailto:mo.tu-bs.de)

In the following, we provide an overview of the notation, more details on the methods used in the main paper along with additional results for the presented algorithm. We further give additional plots displaying more information, such as achieved irradiance and surface errors. This supplementary material is mainly provided for the convenience of the reader, it has been designed to accompany the paper with the purpose of making the work more self-contained.

## Contents

<b>1. Notation</b>	<b>2</b>
1.0.1 Simulation . . . . .	2
1.0.2 Reconstruction . . . . .	2
<b>2. Methods</b>	<b>3</b>
2.1. Derivations Regarding the Pixel-Based Landweber Approach M2 . . . . .	3
<b>3. Result Details</b>	<b>4</b>
3.1. Parameters of All Numerical Experiments . . . . .	4
3.2. Experimental Set-Ups . . . . .	4
3.3. Discussion of Parameters . . . . .	5
<b>4. Additional Results</b>	<b>7</b>
4.1. Print Lines with 5% and 20% Noise Level . . . . .	7
4.1.1 20% Noise Level . . . . .	7
4.2. Comparison to CD (caustic design) of Schwartzburg et al. [12] . . . . .	14
4.3. Complex Set-Up: Two Simple Lenses . . . . .	16
4.4. More Complex Set-Ups: Prism and Lens Array . . . . .	17

# 1. Notation

## 1.0.1 Simulation

$\ell = (\ell_x, \ell_y, \ell_z)^\top$	Position of point light (sensor is in x-y plane).
$h \in \mathbb{R}$	Height of glass substrate.
$d \in \mathbb{R}^{m \times m}$	Height field of 3D-printed glass.
$m$	Height field resolution.
$b \in \mathbb{R}_+^{c \times n \times n}$	Irradiance/Brightness of caustic sensor.
$s$	Distance of the sensor to the substrate.
$c$	Number of wavelengths/channels.
$n$	Photon map resolution.
$F_{h,s}$	Forward operator (simulation): $F_{h,s}: \mathbb{R}^{m \times m} \rightarrow \mathbb{R}_+^{c \times n \times n}$ .
$N_r$	Number of photons for reference generation.
$N_o$	Number of photons for each reconstruction iteration.
$E$	Operator computing the irradiance at the receiver plane.
$x \in \mathbb{R}^3$	Position on the receiver plane.
$x_P \in \mathbb{R}^3$	Position of intersection of photon $P$ with the receiver plane.
$M_P$	Change of basis matrix which transforms the elliptical into a spherical footprint.
$K$	Photon splating kernel.
$E_P$	Irradiance of Photon $P$ .
$P_{h,s}$	Path tracing operator.

## 1.0.2 Reconstruction

$b^\delta \in \mathbb{R}_+^{c \times n \times n}$	Brightness with noise (data).
$\delta > 0$	Relative noise level, i.e. $\ b^\delta - b\ _F \leq \delta \ b\ _F$ .
$\ \cdot\ _F$	Frobenius norm.
$\tau_{\text{dis}}$	Tolerance parameter of discrepancy principle.
$\tau_p > 0$	Step size of naive approach and pixel-based Landweber
$\alpha_p > 0$	Regularization parameter to enforce sparsity in pixel basis.
$\mathbb{S}$	Soft-shrinkage operator.
$p_1$	Lower physical bound of the height field $d$ .
$p_2$	Upper physical bound of the height field $d$ .
$\mathcal{I}_{[p_1, p_2]}$	Interval projection operator.
$\tau_w > 0$	Step size of wavelet-based Landweber.
$\alpha_w > 0$	Regularization parameter to enforce sparsity in wavelet basis.
$W$	Discrete wavelet transform.
$W^{-1}$	Inverse of discrete wavelet transform.
$V$	Volume of the glass.
$V_\epsilon$	Relative uncertainty of the volume of the glass used for printing.
$a$	Area of a pixel (on the glass substrate).
$v$	Volume of reconstructed height field divided by $a$ .
$q_1$	Lower bound for volume heuristic.
$q_2$	Upper bound for volume heuristic.
$j$	$j = (j_1, j_2)$ , indices of height field $d$ .
$r_V$	Surrounding pixels in this radius are included in volume heuristic.
$\bar{d}$	Mean value.
$\check{d}$	Printed part of height field $d$ .
$\dot{d}$	Unprinted part of height field $d$ .
$\check{N}$	Indices corresponding to printed part.
$\dot{N}$	Indices corresponding to unprinted part.
$\gamma$	Parameter to influence the growing of the volume (for volume heuristic).

## 2. Methods

### 2.1. Derivations Regarding the Pixel-Based Landweber Approach M2

As described in the paper, we have to solve the minimization problem

$$\min_{d \in \mathbb{R}^{m \times m}} \frac{1}{2} \|F_{h,s}(d) - b^\delta\|_{\mathbb{F}}^2 + \alpha_p \|d\|_1. \quad (1)$$

This can be solved with the well-known thresholded, nonlinear Landweber scheme, see, e.g., [1, 3],

$$d^{[n+1]} = \mathbb{S} \left( d^{[n]} - \tau_p^{[n]} \nabla f_{\text{dis,p}}(d^{[n]}), \tau_p^{[n]} \alpha_p \right), \quad (2)$$

**Derivation of the Landweber Scheme** In the following we repeat the idea of the derivation of the Landweber scheme and introduce required notation. The question is, when zero is an element of the subdifferential of (1). First,  $\partial f_{\text{dis,p}}(d) = \partial \left\{ \frac{1}{2} \|F_{h,s}(d) - b^\delta\|_{\mathbb{F}}^2 \right\} = [F'_{h,s}(d)]^* [F(d) - b^\delta]$  (compute the subdifferential of the norm written as inner product and change the positions using the adjoint) and second,  $\partial \{ \alpha_p \|d(x)\|_1 \} = \alpha_p \text{sign}(d(x))$  with

$$\text{sign}(d(x)) = \begin{cases} \{-1\} & \text{if } d(x) < 0, \\ [-1, 1] & \text{if } d(x) = 0, \\ \{1\} & \text{if } d(x) > 0, \end{cases} \quad x \in \mathbb{R}^{m \times m}.$$

Note that the subdifferential of the second term coincides with the subdifferential of the absolute value function, i.e. the set-valued signum function.

We consider  $0 \in \{ [F'_{h,s}(d)]^* [F_{h,s}(d) - b^\delta] + \alpha_p \text{sign}(d) \}$ , subtract  $[F'_{h,s}(d)]^* [F(d) - b^\delta]$ , multiply with step size  $\tau_p > 0$  and add  $d$ . This results in  $d - \tau_p [F'_{h,s}(d)]^* [F_{h,s}(d) - b^\delta] \in (I + \tau_p \alpha_p \text{sign})(d)$ . The inverse of  $(I + \tau_p \alpha_p \text{sign})$ , i.e., the proximal mapping  $(I + \tau_p \alpha_p \text{sign})^{-1}(d)$  is the already introduced soft-shrinkage operator with  $\kappa = \tau_p \alpha_p$ . Finally, we end up with (2).

**Derivation of the Extended Landweber Scheme** The derivation of the extended Landweber scheme is quite similar to the non-extended original. In addition to (1), we have to take into account the subdifferential of  $\delta_{[p_1, p_2]}(d)$ .

To this end, the auxiliary, set-valued step function  $\text{step}_{[p_1, p_2]}$  and their inverse  $\text{step}_{[p_1, p_2]}^{-1}$  defined by

$$\text{step}_{[p_1, p_2]}(x) := \begin{cases} \{p_1\} & \text{if } x \leq 0, \\ [p_1, p_2] & \text{if } x = 0, \\ \{p_2\} & \text{if } x \geq 0, \end{cases}$$

and

$$\text{step}_{[p_1, p_2]}^{-1}(x) := \begin{cases} (-\infty, 0] & \text{if } x = p_1, \\ \{0\} & \text{if } x \in (p_1, p_2), \\ [0, \infty) & \text{if } x = p_2, \\ \emptyset & \text{otherwise,} \end{cases}$$

turn out to be useful. Of course, the element-wise application is assumed if  $x$  is a vector. Note that  $\text{step}_{[-1, 1]}(d) = \text{sign}(d)$  and  $\text{step}_{[p_1, p_2]}^{-1}(d) = \partial \delta_{[p_1, p_2]}(d)$ .

Therefore,  $(I + \tau_p \partial \{ \alpha_p \|\cdot\|_1 + \delta_{[p_1, p_2]}(\cdot) \})(d)$  is the same as  $d + \tau_p \alpha_p \text{step}_{[-1, 1]}(d) + \text{step}_{[p_1, p_2]}^{-1}(d)$ . The inverse of this term exists and is called extended soft-shrinkage operator  $\mathcal{I}_{[p_1, p_2]}(\mathbb{S}(d, \tau_p \alpha_p))$ .

### 3. Result Details

#### 3.1. Parameters of All Numerical Experiments

Lens- es	Ground truth	s (cm)	#wave- lengths	Noise level	Recon. method	Fig.	Run time (min)	Rel. dis.	Rel. err.	#iter	$\tau_{dis}$	$\tau_p$	$\alpha_p$	$[p_1, p_2]$	$\tau_w$	$\alpha_w$	$V_\varepsilon$	$\gamma$
Print lines with 5% noise level, see Figs. 4–9																		
N-N	<i>Lines</i>	0	1	5%	M2V1	4(b)	2.95	.0504	.0648	72	1.01	.1	.002	[0, 0.3]	.1	.01	.25	.1
					M3V1	4(c)	2.77	.0505	.0658	65	1.01	.1	.0005	[0, 0.3]				
					M1	4(d)	6.62	.0549	.1800	162	1.1	.1						
					M2V0	4(e)	4.72	.0600	.1086	114	1.2	.1	.002	[0, 0.3]				
					M3V0	4(f)	4.38	.0598	.1065	102	1.2	.1	.0005	[0, 0.3]				
Print lines with 20% noise level (in addition to the paper), see Figs. 4–9																		
N-N	<i>Lines</i>	0	1	20%	M1	4(g)	1.28	.2020	.2007	31	1.01	.1			.1	.01	.25	.1
					M2V1	4(h)	0.58	.1991	.1261	14	1.001	.1	.002	[0, 0.3]				
			M3V1		4(i)	0.62	.1995	.1297	14	1.001	.1	.0005	[0, 0.3]					
			3		M1	4(j)	11.82	.2019	.2028	124	1.01	.005						
					M2V1	4(k)	2.98	.2000	.1298	33	1.001	.005	.002	[0, 0.3]				
M3V1	4(l)	3.05	.2002	.1305	33	1.001	.005	.0005	[0, 0.3]									
Comparison to CD (caustic design) of Schwartzburg et al. [12], see Figs. 10 and 11																		
N-N	<i>Lines</i>	0	1	5%	M2V1		2.95	.0504	.0648	72	1.01	.1	.002	[0, 0.3]	.1	.01	.25	.1
	<i>Lines</i>	0.0125			CD		30.80 <sup>a</sup>	.0693	1.0217									
	A	0.0125			M2V1		2.05	.0548	.1773	34	1.1	.1	.002	[0, 0.3]				
	A	0.0125			CD		31.72 <sup>a</sup>	.0862	.5596									
Two simple lenses, see Fig. 12																		
L-L	A	2.500	1	5%	M2V1	12	8.55 <sup>b</sup>	.0599	.1713	108	1.2	.05	.002	[0, 0.3]	.25	.1		
More complex set-ups: prism and lens array (in addition to the paper), see Figs. 13–16																		
N-A	dam. A	1.875	1	5%	M2V1	13	3.97 <sup>b</sup>	.0747	.2469	54	1.5	.05	.002	[0, 0.15]	.25	.1		
N-A	dam. A <sup>c</sup>	1.875			M2V1	14	3.40 <sup>b</sup>	.0600	.1969	43	1.2	.05	.002	[0, 0.15]	.25	.1		
P-N	dam. A	0.0125			M2V1	15	21.93 <sup>b</sup>	.0749	.2618	242	1.5	.05	.002	[0, 0.15]	.25	.1		
P-N	dam. A <sup>c</sup>	0.0125			M2V1	16	1.85 <sup>b</sup>	.0993	.1885	23	2.0	.05	.002	[0, 0.15]	.25	.1		
Real-world sample																		
N-N	unk. <sup>d</sup>	0	1	unk.	M2V1		1.78	.1740	unk.	30	1.01	.00001	7	[0, 0.035]	.25	10		

Table 1: Run times, relative discrepancies, and errors as well as chosen parameters for all numerical experiments.

All reconstructions were stopped by discrepancy principle if not otherwise stated.

Legend for the lenses: First, we mention the top lens and second, the bottom lens (N: no lens, L: simple lens, P: prism, A: lens array).

Note that the simple top lens is different from the simple bottom lens as described in the experimental set-up in the paper.

In all cases of used volume heuristic, we set the radius  $r_V$  to include surrounding pixels to 2.

<sup>a</sup>Mixed CPU and GPU algorithm, run time not directly comparable.

<sup>b</sup>As a consequence of the additional intersection events, we had to adapt the simulation for the reconstruction using 16 inner simulations (instead of 32) and a splat smoothing parameter of 177 (we divide the default value of 250 by  $\sqrt{2}$ ).

<sup>c</sup>Instead of initial guess for height field  $d = 0$  we use A.

<sup>d</sup>In this experimental set-up we consider a glass substrate of physical dimensions 12.36 mm  $\times$  12.36 mm  $\times$  4.08 mm with a printed line of height 0.12 mm, we place our light source at  $\ell = (0 \text{ cm}, 0 \text{ cm}, 151.36 \text{ mm})^T$  and we use a wavelength of 0.525  $\mu\text{m}$ .

#### 3.2. Experimental Set-Ups

Unless otherwise noted, our experiments share a common set-up: The first common variable is the predefined height field, that consists either of *print lines* or of the character A (and a part of a ring). In brief, we speak of the *lines* or the A. The *lines* are depicted in Fig. 1(a)/(b). The A is shown in Fig. 2. A special case is A *damaged* that is A with some built-in errors. The synthetic height field of print lines was generated to deal with several problems in glass 3D printing like interruptions in the printing line and lines of different thickness. Furthermore, the thinnest print line is a challenge for the resolution of the height field  $d \in \mathbb{R}^{m \times m}$  with  $m = 128$ . In the case of the A, we deal with sharp edges.

The height field is thus assumed to be square and placed on a substrate of physical dimensions 5 cm  $\times$  5 cm  $\times$  0.25 cm. We place our light source at  $\ell = (0 \text{ cm}, 0 \text{ cm}, 25 \text{ cm})^T$  and render our reference solution with  $N_r \approx 3 \cdot 10^7$  photons per

wavelength. Due to memory restrictions in the gradient computation we cannot reconstruct with this high number of photons, thus we use  $N_o \approx 8 \cdot 10^6$  photons per wavelength, taking care to choose the smoothing parameter [2] such that initial differential vectors of photons keep the same length in reference generation and reconstruction phase. For metrological reasons we have chosen typical wavelengths of a helium-neon laser, i.e., wavelengths of 0.633, 1.152 and 3.392  $\mu\text{m}$  as they have the largest gain factors, see [9]. As for the refractive index curve of the glass, we use the fused silica curve from [4, 5, 13] since it includes the infrared and visible spectrum. Unless otherwise noted, we use one wavelength of 0.633  $\mu\text{m}$ .

Unless otherwise stated, the experiments do not include additional optical elements (see Tab. 1 for variations with simple lenses, prism or lens array). The corresponding caustic images of all three wavelengths in the case of the print lines are shown in Fig. 1(c). The caustic image of one wavelength in the case of the *A* is depicted in Fig. 2(c).

By default, Gaussian noise with noise level  $\delta$  of 5% was employed to simulate data. Deviating from this, we have also examples with 20%, see Tab. 1.

However, we perturb the data with slightly less than 5% or 20% as the forward operator used for reconstruction process has an intrinsic error of approximately 0.4% in the case of one wavelength and 0.6% in the case of three wavelengths, meaning that only the difference to the desired noise level will be added.

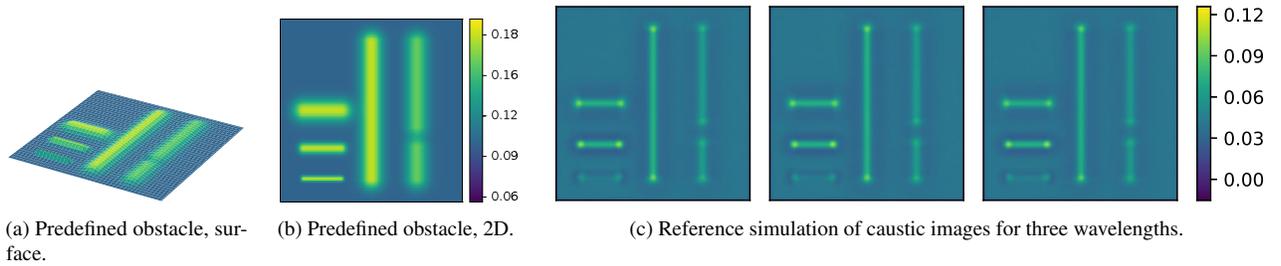


Figure 1: Simulation results of *lines*. Surface (a) and 2D (b) views of the predefined obstacle (height field  $d$  of printed glass on top of glass substrate). The simulation of the corresponding caustic images, i.e., irradiance at sensor, for the three wavelengths of 0.633, 1.152 and 3.392  $\mu\text{m}$  are depicted in (c).

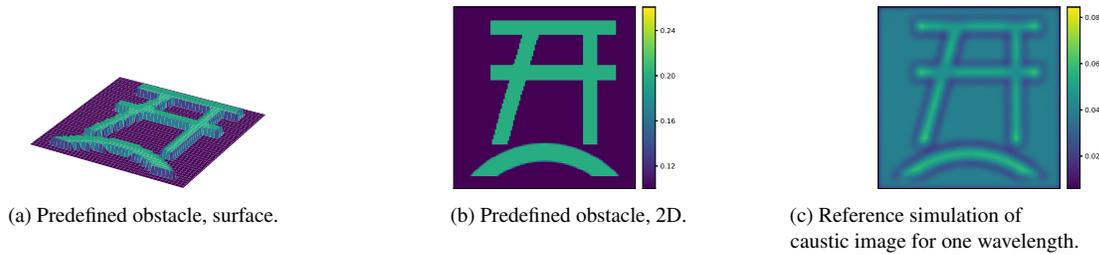


Figure 2: Simulation results of *A*. Surface (a) and 2D (b) views of the predefined obstacle (height field  $d$  of printed glass on top of glass substrate). The simulation of the corresponding caustic image, i.e., irradiance at sensor, for the wavelength of 0.633  $\mu\text{m}$  is depicted in (c).

### 3.3. Discussion of Parameters

We discuss the parameters based on the numerical examples of the print lines with 5% and 20% noise level without any additional optical elements. However, these parameters are suitable for most other cases, cf. Tab. 1.

It is known that the naive approach as well as the Landweber scheme are sensitive to the choice of their step size: they

are very slow if the value is chosen too small; the reconstruction fails completely if the value is chosen too big. After several experiments, the value of  $\tau_p = 0.1$  was deemed best for the naive approach as well as the pixel-based Landweber in the case of one wavelength. We use  $\tau_w = \tau_p = 0.1$  for the wavelet-based Landweber in the case of one wavelength. For three wavelengths we adapt the step sizes to  $\tau_p = 0.005$  and  $\tau_w = 0.005$ .

We set the tolerance parameter  $\tau_{\text{dis}}$  to 1.1 for 5% noise level and to 1.01 for 20% noise level if we employ the naive approach. If the pixel-based or wavelet-based Landweber is used and the noise level is 5%, we set  $\tau_{\text{dis}}$  to 1.01 in the case of enabled volume heuristic and to 1.2 in the disabled case. If Landweber with volume heuristic is used in the case of 20% noise level, we set the tolerance parameter  $\tau_{\text{dis}}$  to 1.001.

The regularization parameters were determined manually. A rule of thumb is to set them as high as possible such that the discrepancy principle is fulfilled. We set them independently of the noise level. The regularization parameter  $\alpha_p$  for the sparsity in the pixel basis is set to 0.002 if the pixel-based Landweber is employed. The corresponding parameter in the case of wavelet-based Landweber is  $\alpha_p = 0.0005$  and for the regularization parameter for the sparsity in a wavelet basis we use  $\alpha_w = 0.01$ . As already mentioned we decided on Daubechies 3 wavelets. In both cases of Landweber-based reconstruction the physical bounds are enforced between  $p_1 = 0$  and  $p_2 = 0.3$ .

If the volume heuristic is used, we assume to know the volume of the deposited material with a relative uncertainty of  $V_\varepsilon = 0.25$  in all cases. To influence the growing we set  $\gamma = 0.1$  in the case of one wavelength. For three wavelengths a slower growth turned out to be advantageous,  $\gamma = 0.05$ . In all cases we set the radius  $r_V$  to include surrounding pixels to 2.

## 4. Additional Results

### 4.1. Print Lines with 5% and 20% Noise Level

Based on the experimental set-up described above, we consider three experimental set-ups for the print lines in this section and call them W1N05, W1N20 and W3N20: In the case of W1N05, we use one wavelength of  $0.633 \mu\text{m}$ . Gaussian noise with noise level  $\delta$  of 5% was employed to simulate data in this case. (This default case was already considered in the main paper.) In the case of W1N20, we also use a wavelength of  $0.633 \mu\text{m}$  with  $\delta = 20\%$ . In the case of W3N20, we use all three wavelengths. Again, 20% Gaussian noise was used in the data simulation.

#### 4.1.1 20% Noise Level

All reconstruction results to 20% noise level are in Fig. 4. For a better comparison, the colorbar limits are the same as is in the corresponding figure in the paper and we repeat the results to 5% noise level.

As expected, the reconstruction with the naive approach differs strongly from the predefined one in all experimental set-ups, see Fig. 4(g)/(j). In particular, this is visible in the slices, see Fig. 3(g)/(j). In addition, this method needs the longest run time in all experimental set-ups. A special characteristic is that the reconstructed form is partly below the substrate height of 0.1.

Both pixel-based as well as wavelet-based Landweber, see second and third column of Figs. 4 and 3, deal with that problem as they take into account physical bounds and deliver sensible reconstructions as they enforce sparsity. In addition, the area-based physical bounds ensure a mostly artifact-free background. In our experiments it is not crucial whether we employ a pixel-based or wavelet-based scheme.

We show the robustness of the used pixel-based and wavelet-based Landweber (with all described extensions) in Figs. 4(h)/(i) and 3(h)/(i). We receive a sensible reconstruction in the case of 20% noise level with one wavelength. It takes less than a minute to reach a reconstruction error of 13%.

In the case of three wavelengths (with 20% noise level), the reconstruction benefits from additional information and shows approaches of the thinnest print line, see Figs. 4(k)/(l) and 3(k)/(l). In that case pixel-based and wavelet-based Landweber can outperform the naive approach very clearly in terms of run time by factor four. The reconstruction error is better anyway, i.e. 13%, which is in the order of magnitude of the experiment with one wavelength (and 20% noise level). Therefore, we are confident that our reconstruction algorithm is able to deal with real-world data as well.

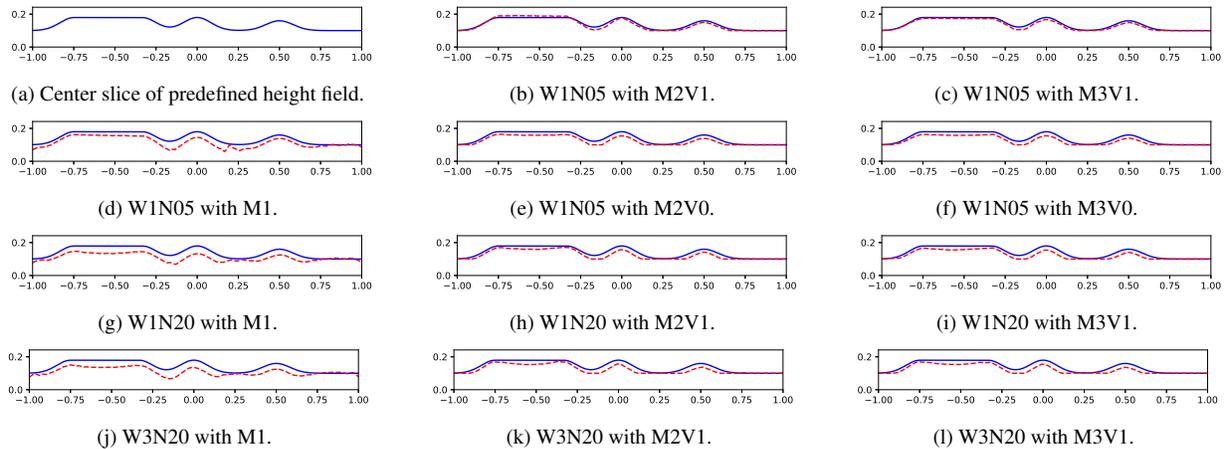


Figure 3: Center slices of the predefined height field (blue solid) and the reconstructions (red dashed) to compare the reconstruction methods. The layout is the same as in Fig. 4. One unit on the x-Axis corresponds to 2.5 cm.

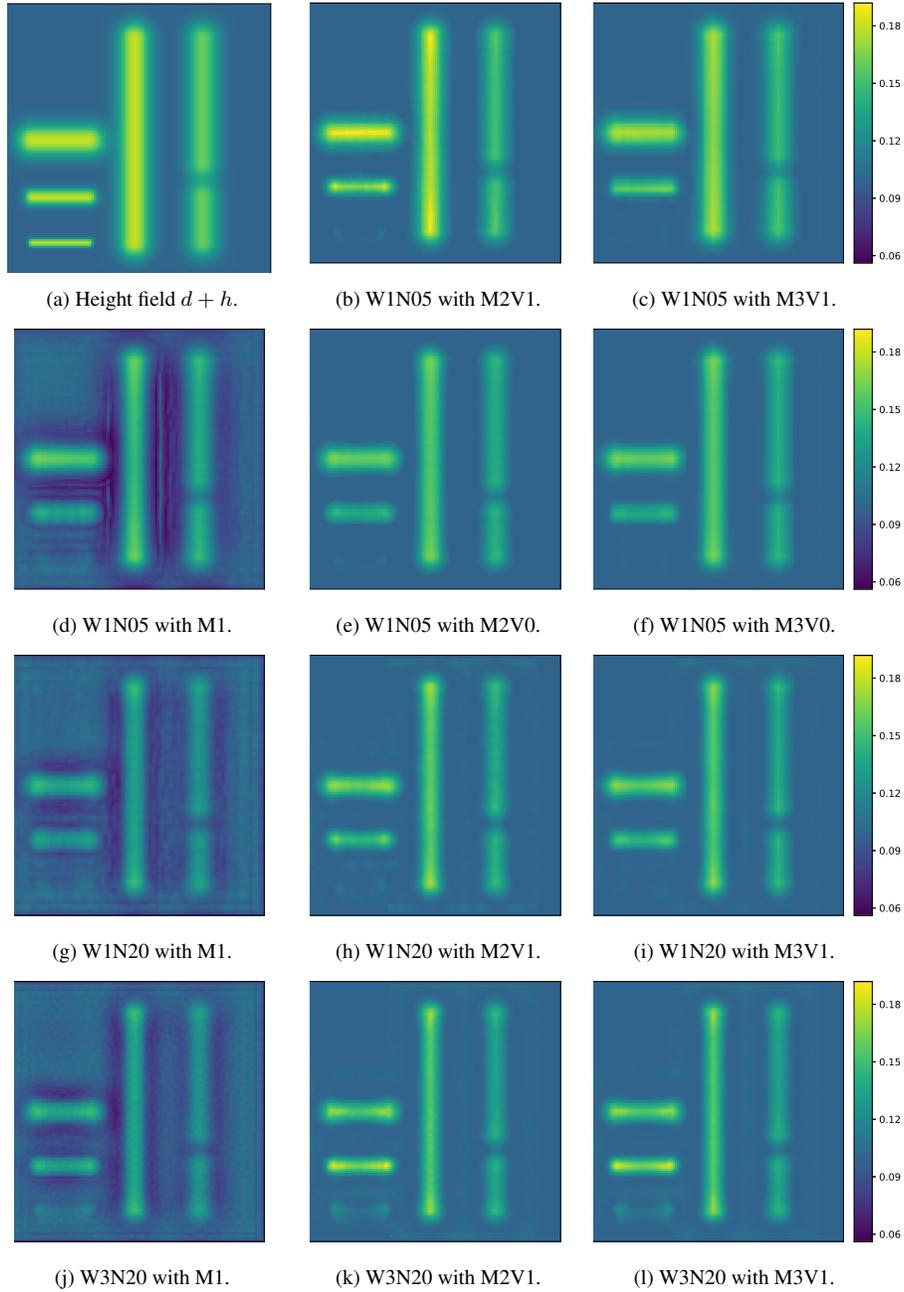


Figure 4: 2D Comparison of reconstruction methods in the case of one (W1) and three wavelengths (W3), see (b)–(i) and (j)–(l), for simulated data with 5% (N05) and 20% noise level (N20), see (b)–(f) and (g)–(l). Apart from the predefined height field (a) the reconstructions are depicted with the naive approach (M1) in the first column, the pixel-based Landweber (M2) in the second one, and the wavelet-based Landweber (M3) in the third one. We demonstrate the effect of the proposed volume heuristic (V1) in (b) and (c) in comparison to disabling (V0) in (e) and (f). In all other cases of the Landweber schemes we employ the volume heuristic. Note that all colorbars have the same limits. The corresponding center slices of the predefined height field and the reconstructions are depicted in Fig. 3 in the same layout. All reconstructions were stopped by discrepancy principle. The run times, relative discrepancies, and relative errors as well as the chosen parameters are indicated in Tab. 1.

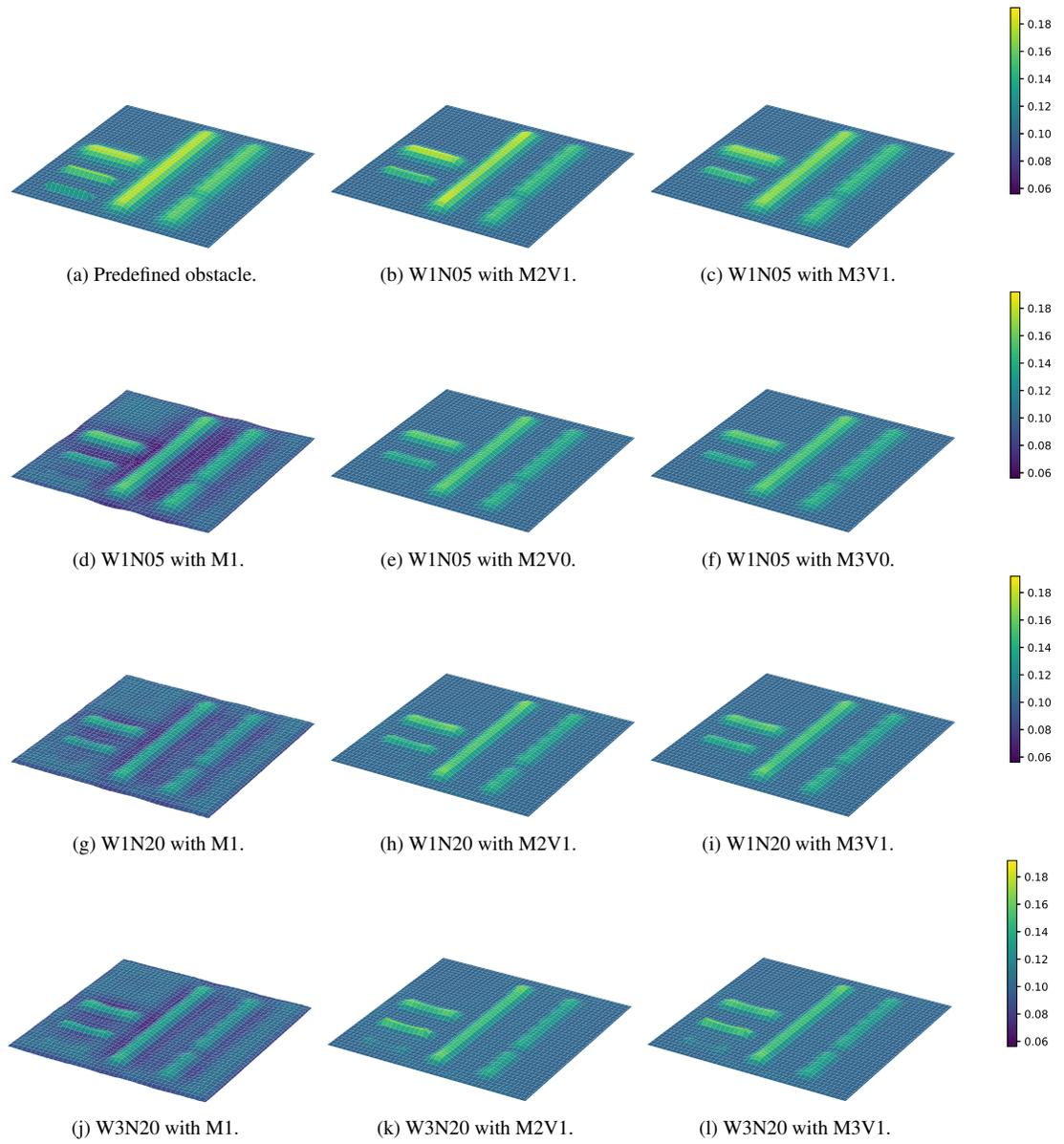


Figure 5: Reconstruction of the obstacle (height field  $d$  on top of a substrate with height  $h$ ) with different methods as surface plot. Layout as in Fig. 4. Note that all colorbars have the same limits.

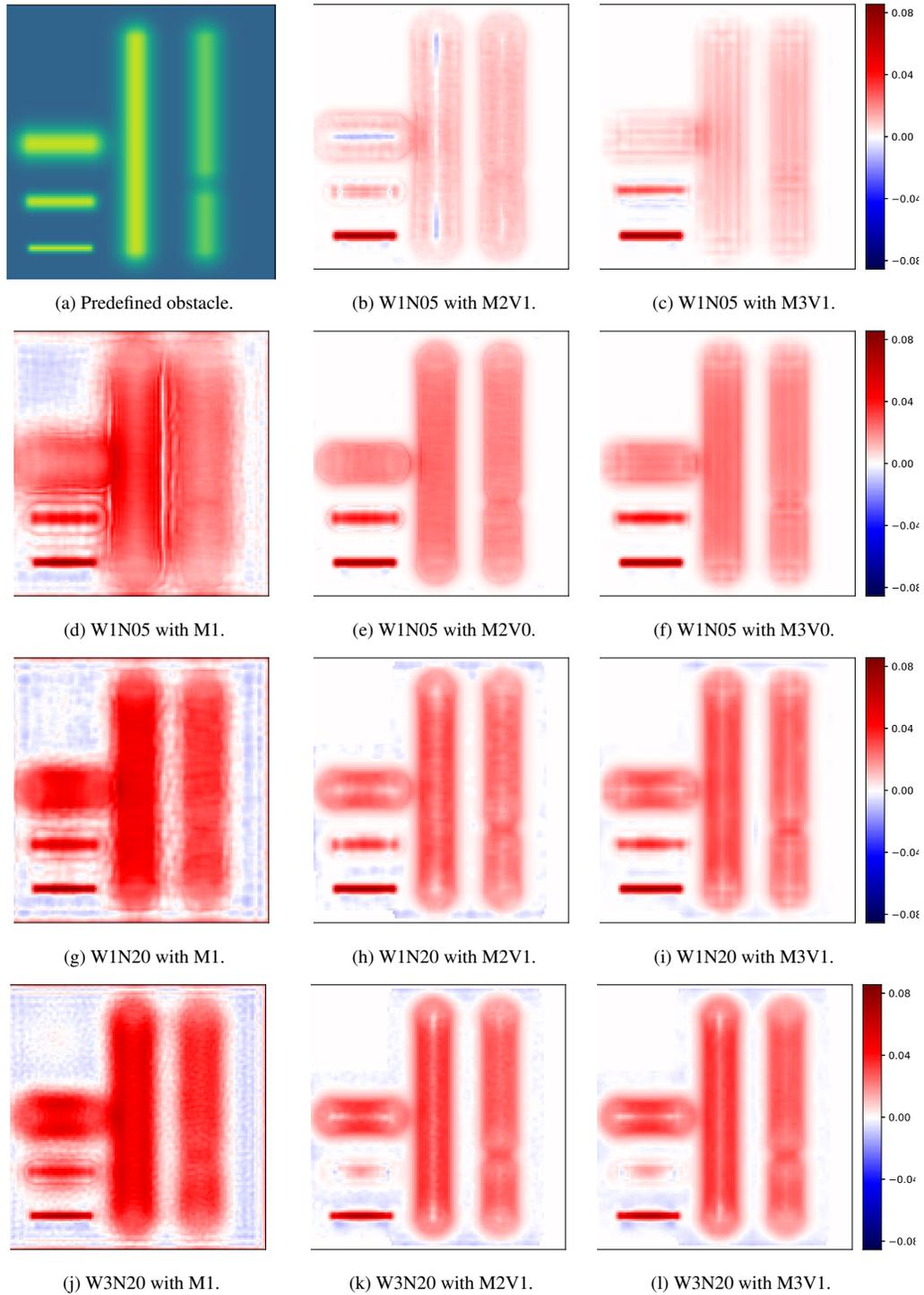


Figure 6: Error of reconstruction (predefined and reconstructed obstacle) as 2D plot. Layout as in Fig. 4. Note that all colorbars except the first have the same limits.

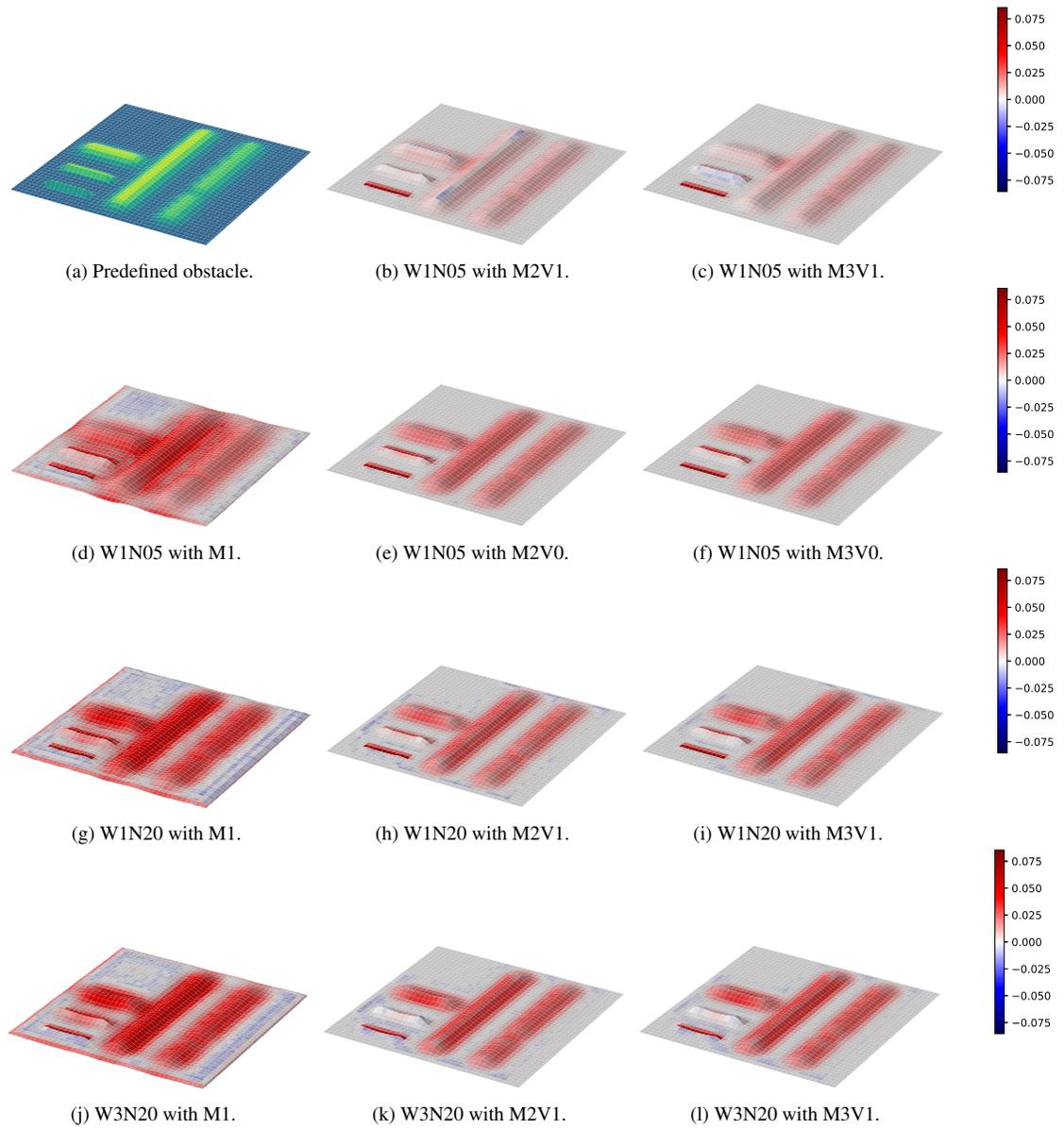


Figure 7: Error of reconstruction (predefined and reconstructed obstacle) as surface plot. Layout as in Fig. 4. Note that all colorbars except the first have the same limits.

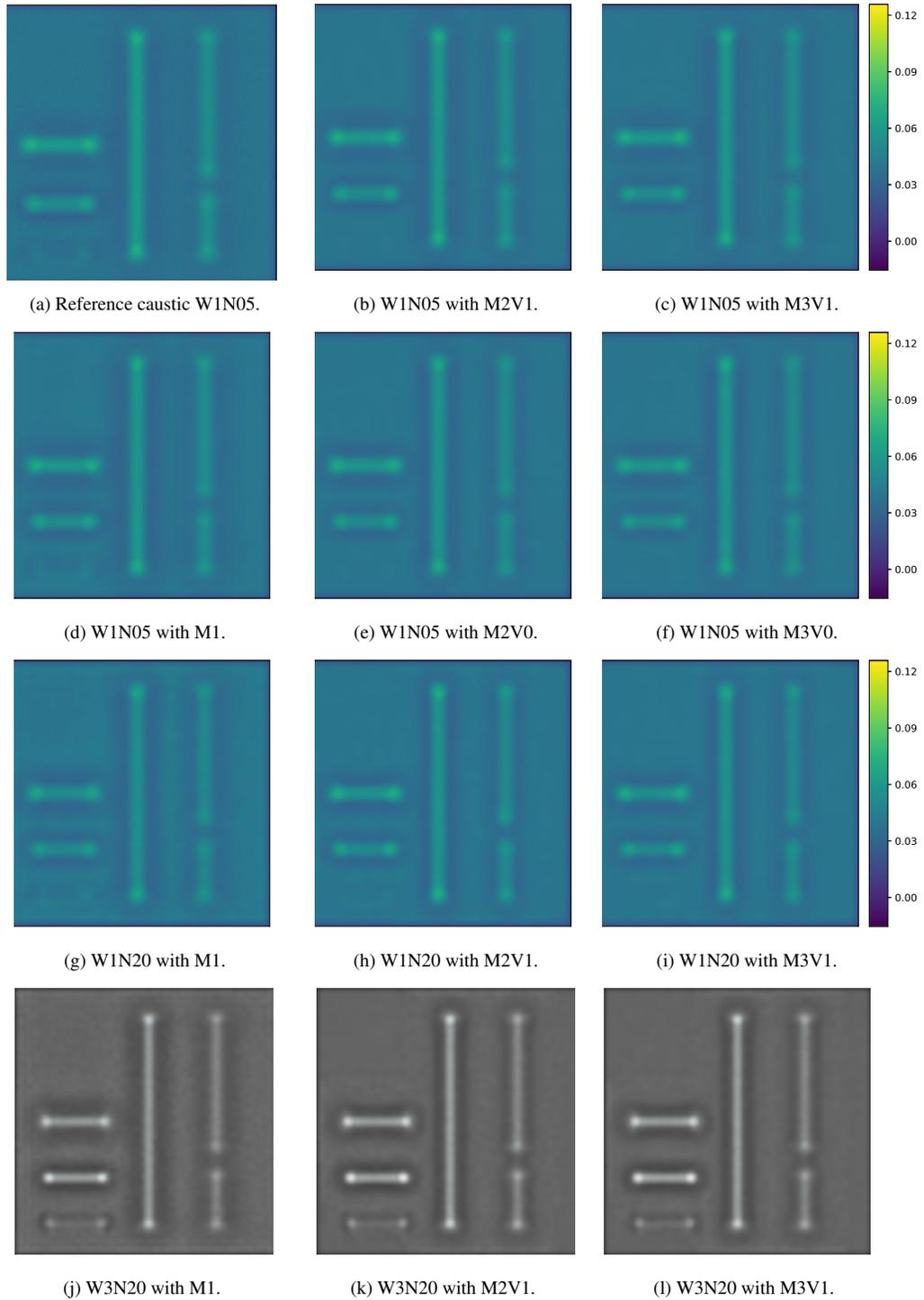


Figure 8: Predicted caustic (from reconstruction results) from different methods in comparison to reference caustic in 2D plots. Layout as in Fig. 4. Note that all colorbars have the same limits and multi-spectral images have been mapped to RGB channels.

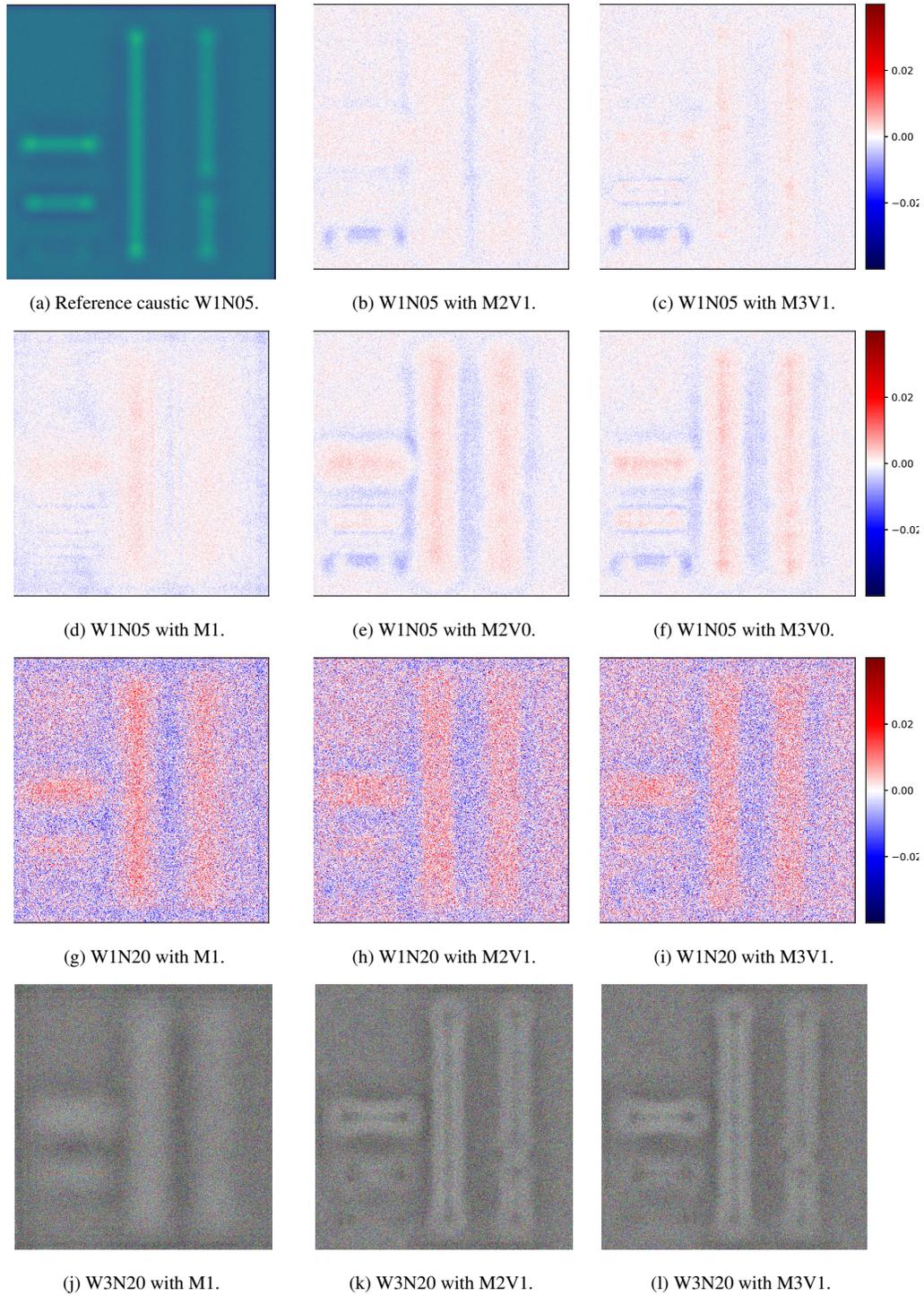


Figure 9: Error of predicted caustic (predicted and reference caustic) of different methods as surface plot. Layout as in Fig. 4. Note that all colorbars except the first have the same limits and multi-spectral images have been mapped to RGB channels.

## 4.2. Comparison to CD (caustic design) of Schwartzburg et al. [12]

In this section we will provide additional details for our implementation of Schwartzburg et al. [12] including the utilized parameters and how we compare to our method: Since we were unable to find an official baseline implementation of Schwartzburg’s method, we chose to reimplement it, based on the description in the paper. Our implementation differs in some aspects from the original and some parameters had to be hand-tuned again, both of which we will detail in the following:

For computation of the optimal transport map, we use the original implementation of [6, 7] and its Python wrapper [8]. Our highest resolution in the multiscale hierarchy consists of  $2^{15}$  sites. As in the original we divide the number of points by 4 for each level of the hierarchy. We further use 100 iterations of Lloyds sampling to smooth the resulting site positions. In the multiscale optimal transport computation we pass the objective and gradient from [8] into the L-BFGS-B optimizer from [14] and use predefined convergence criteria.

In the target optimization step we assume the same light source, screen position and single wavelength to achieve comparability with our method. We compute the target mesh with  $287^2$  vertices. In contrast to the original paper, we implement the energy function for target optimization in PyTorch [10, 11] to use its autodifferentiation and GPU-acceleration capabilities. As such we use the L-BFGS optimizer from PyTorch. We stop the optimization, when the losses plateau (i.e.  $\|f_{k+1} - f_k\| \leq 10^{-5}$ ) or when the target shape does not change anymore (i.e.  $\|x_{k+1} - x_k\| \leq 10^{-5}$ ). Regarding the outer iteration in the target optimization step in which a new target normal field is computed, we use a similar convergence criterion: We stop, when the target shape does not change significantly, i.e.  $\|x_{k+1} - x_k\| \leq 10^{-5}$ .

Finally we set the barrier energy to penalize vertices, which fall under the initial substrate height, as we are only interested in solutions arising from the addition as material. We set the weighting parameters as follows: the weight for the integration energy ( $E_{int}$ ) is set to 1 as well as the weight for the barrier energy ( $E_{bar}$ ). The direction energy ( $E_{dir}$ ) and regularization energy ( $E_{bar}$ ) however, are weighted with  $10^5$ . We do not use the flux energy ( $E_{flux}$ ) in our examples.

The target optimization gives a result as a mesh with vertices, which may vary not only in height. To compare this result to our height field representation we interpolate a height field from the resulting mesh at our GT height field resolution. We have found that the resulting meshes are sufficiently regular for the feature size under consideration here.

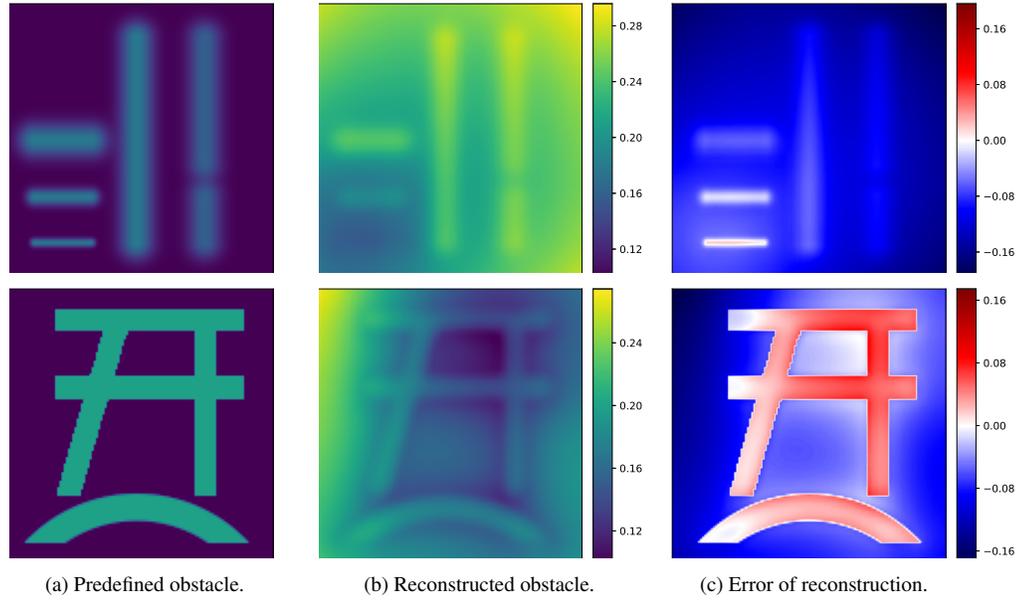


Figure 10: Additional plots for caustic design [12] reconstructions of the predefined obstacle on the *lines* (top) and *A* (bottom) example. Note that the obstacles share the same colorbar.

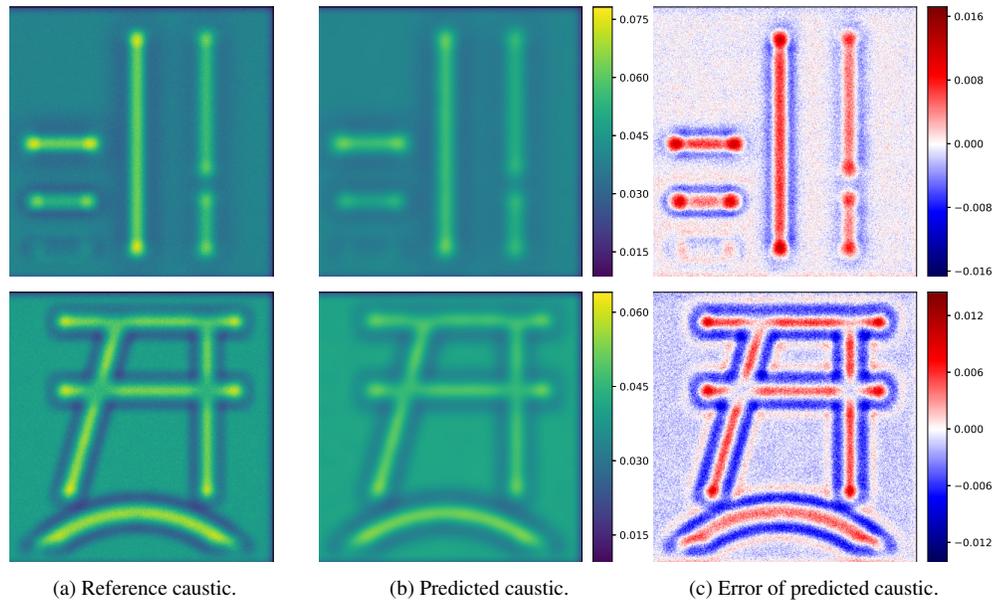


Figure 11: Additional plots for caustic design [12] reconstructions of the resulting caustic image on the *lines* (top) and *A* (bottom) example. Note that the caustic images share the same colorbar.

### 4.3. Complex Set-Up: Two Simple Lenses

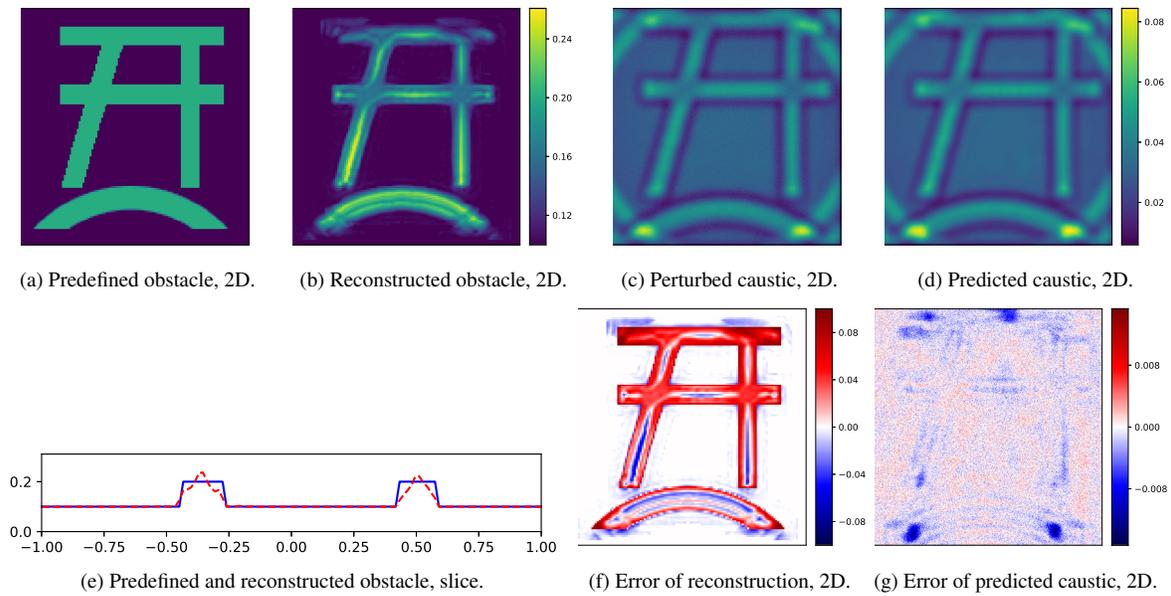


Figure 12: Reconstruction of an obstacle (consisting of  $A$  on top of a substrate) between two simple lenses with our method M2V1.

#### 4.4. More Complex Set-Ups: Prism and Lens Array

We want to consider the limits of our reconstruction method M2V1 with experimental set-ups consisting of a lens array or a prism. In both cases the 3D printer should print the  $A$  obstacle. However, we assume that a damaged version is printed. So, the predefined obstacle is a damaged  $A$  obstacle. First, we compute a reconstruction as in the numerical experiments before with an initial guess of the height field  $d = 0$ , see Figs. 13 and 15. Second, we start as initial guess with the height field of the  $A$  without damage, see Figs. 14 and 16.

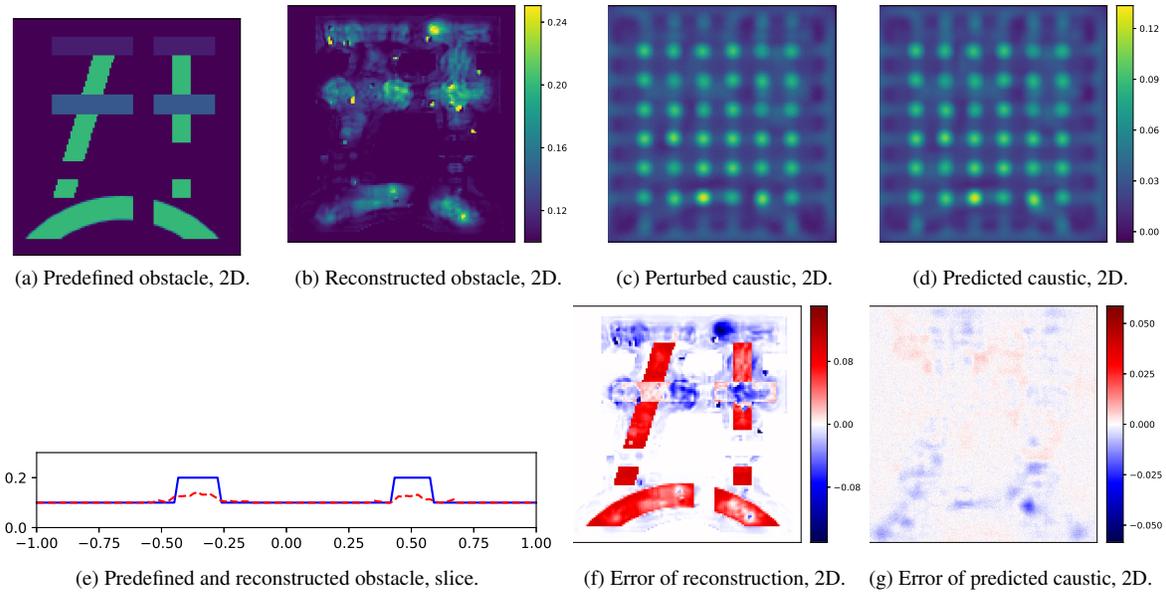


Figure 13: Reconstruction of an obstacle (consisting of damaged  $A$  on top of a substrate) with our method M2V1 in the case of a lens array as bottom optical element.

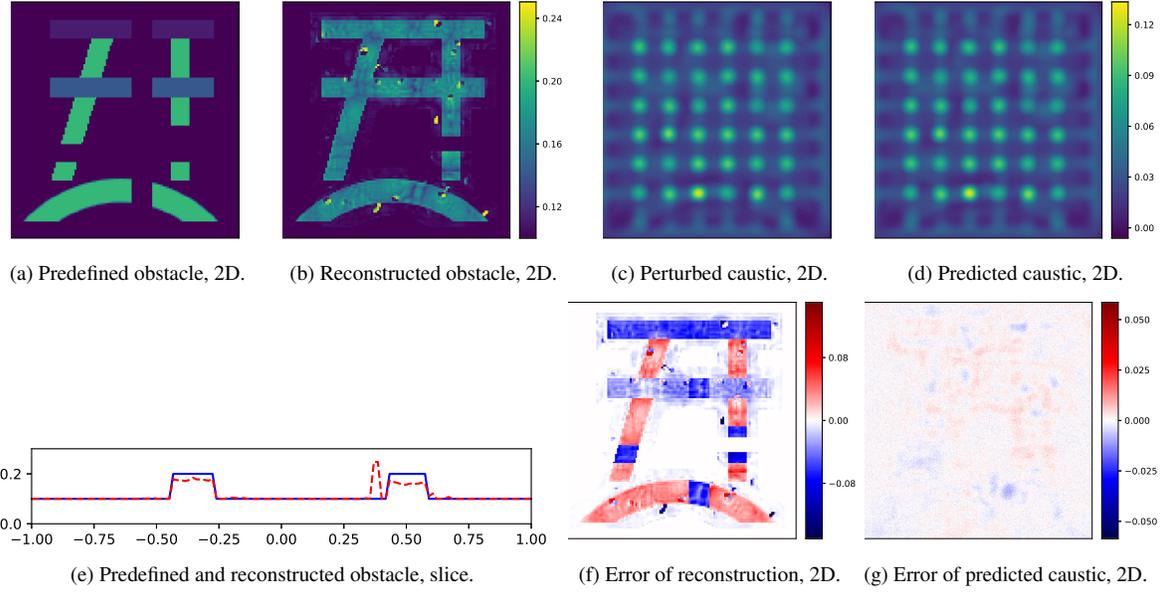


Figure 14: Reconstruction of an obstacle (consisting of damaged  $A$  on top of a substrate) with our method M2V1 in the case of a lens array as bottom optical element. As initial guess we employ  $A$ .

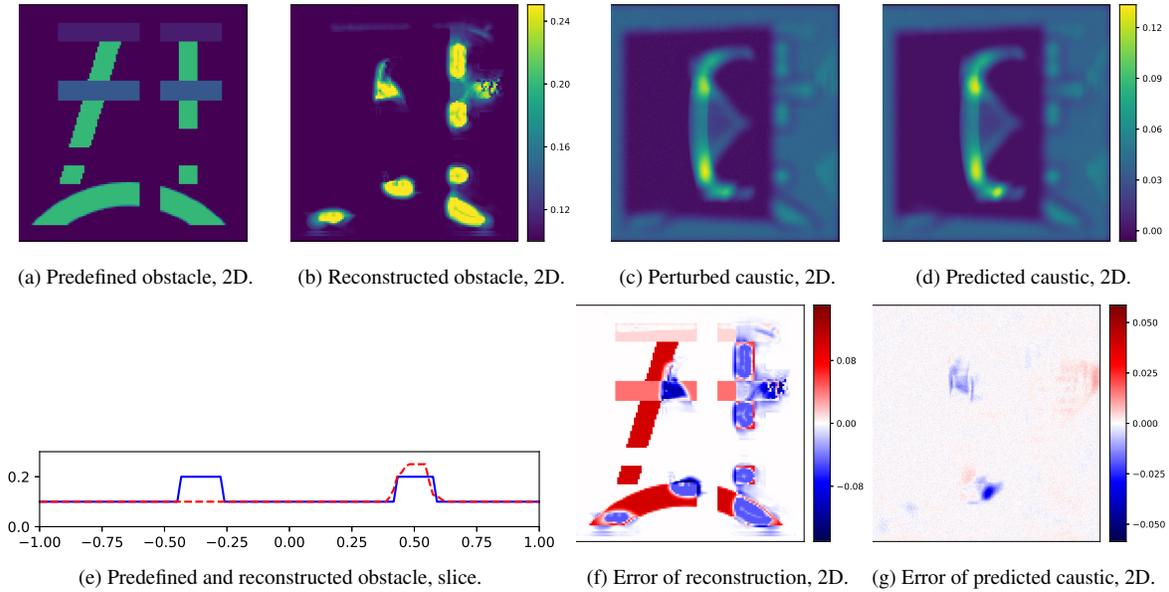


Figure 15: Reconstruction of an obstacle (consisting of damaged  $A$  on top of a substrate) with our method M2V1 in the case of a prism as top optical element.

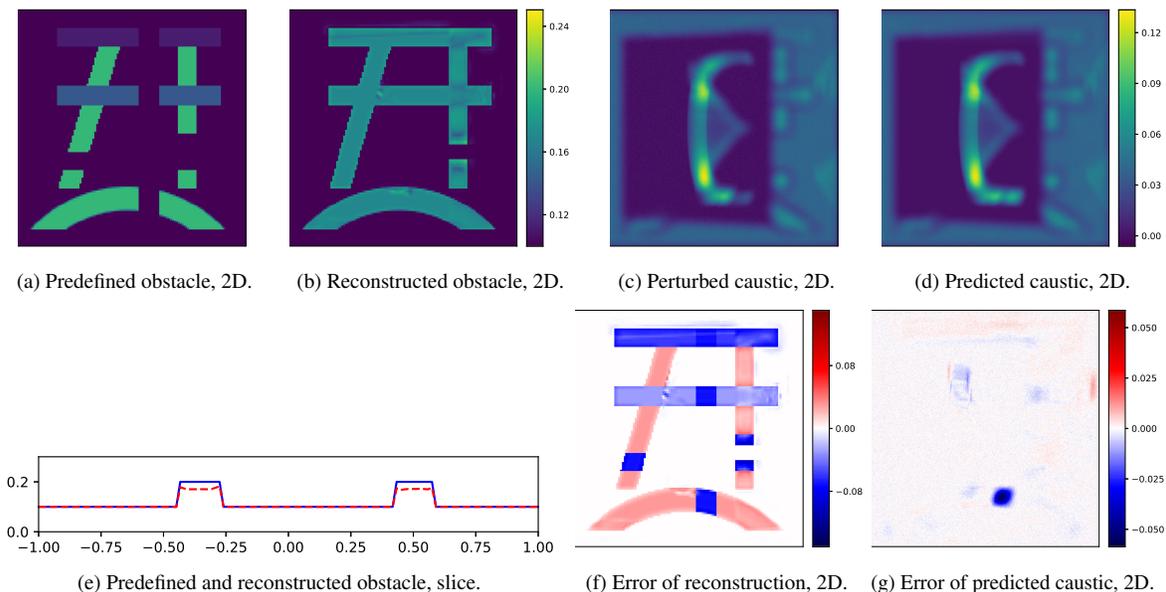


Figure 16: Reconstruction of an obstacle (consisting of damaged  $A$  on top of a substrate) with our method M2V1 in the case of a prism as top optical element. As initial guess we employ  $A$ .

## References

- [1] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.
- [2] J.R. Frisvad, L. Schjøth, K. Erleben, and J. Sporring. Photon differential splatting for rendering caustics. *Computer Graphics Forum*, 33(6):252–263, Sept. 2014.
- [3] A. Lechleiter, K.S. Kazimierski, and Mirza Karamehmedović. Tikhonov regularization in  $L^p$  applied to inverse medium scattering. *Inverse Problems*, 29(7):075003, 2013.
- [4] I.H. Malitson and C. Tan. Optical constants of fused silica (fused quartz), Apr. 2020. [Online; Last accessed 17-April-2020].
- [5] I. H. Malitson. Interspecimen comparison of the refractive index of fused silica. *Josa*, 55(10):1205–1209, 1965.
- [6] Quentin Mérigot. A multiscale approach to optimal transport. *Comput. Graph. Forum*, 30(5):1583–1592, Aug. 2011.
- [7] Quentin Mérigot. Mongeampere++. <https://github.com/mrgt/MongeAmpere>, 2018. Accessed: 2020-07-13.
- [8] Quentin Mérigot. Pymongeampere. <https://github.com/mrgt/PyMongeAmpere>, 2019. Accessed: 2020-07-13.
- [9] D. Meschede. *Optics, Light, and Lasers*. John Wiley & Sons, Ltd, 2017.
- [10] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NIPS 2017 Workshop on Autodiff*, 2017.
- [11] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
- [12] Y. Schwartzburg, R. Testuz, A. Tagliasacchi, and others. High-contrast computational caustic design. *ACM Transactions on Graphics*, 2014.
- [13] CZ Tan. Determination of refractive index of silica glass for infrared wavelengths by ir spectroscopy. *Journal of Non-Crystalline Solids*, 223(1-2):158–163, 1998.

- [14] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.