# Mask-ToF: Learning Microlens Masks for Flying Pixel Correction in Time-of-Flight Imaging Supplemental Document

In this Supplemental Document, we hope to further elucidate points brought up in the main text by providing supporting details, results, and discussion thereof. Specifically, we provide

- Intuition behind spatial multiplexing and the Flying-Pixel/Signal-to-Noise-Ratio tradeoff (Section 1)
- Detailed discussion of mask patterns and their evolution (Section 2)
- Additional details on mask fabrication and optical relay design (Section 3)
- Discussion of error sources in experimental data (Section 4)
- Additional details concerning network design and implementation (Section 5)
- Additional results for synthetic data (Section 6)
- Additional results for experimental data (Section 7)

## 1. The Flying-Pixel/Signal-to-Noise-Ratio Tradeoff

In this section, we discuss the fundamental tradeoff between signal-to-noise ratio (SNR) and flying pixel (FP) count imposed by a global aperture system, and how Mask-ToF circumvents this limitation.

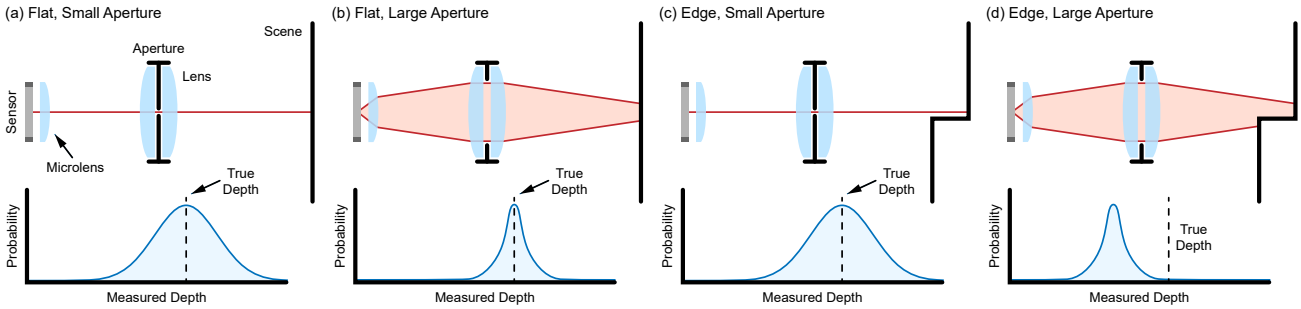


Figure 1: Visualization of the FP/SNR tradeoff. We illustrate the light envelope of a single sensor pixel with pinhole (a, c) and wide aperture (b, d) setups and for locally flat (a, b) and discontinuous (c, d) scenes. We note that in all cases the focal plane is behind the scene. The plots underneath each setup depict the expected distribution of depth measurements — perturbed by system and environmental noise — as recorded by the sensor pixel. As in the main text, we consider the depth at the chief ray to be the *true* depth.

In general, increasing aperture size allows for the collection of more light, reducing the effects of system noise and photon shot on the sensor's measured depth as more light paths are averaged. We correspondingly see that the distributions of

measurements for wide aperture setups (Figure 1 (b,d)) are significantly narrower than those of the pinhole setups (Figure 1 (a,c)). This demonstrates one side of the tradeoff, that signal-to-noise ratio (SNR) is expected to increase as aperture size increases, and vice versa. For locally flat scenes (Figure 1 (a,b)) — ones without an object discontinuity in view of the sensor pixel — the aperture integrates light paths of near-identical length. This means the expected distribution of measurements does not shift as more and more light is let into the sensor. In this case, a wide aperture is a net positive, as it narrows the distribution, on average bringing the measurements closer to the true depth.

This is not necessarily the case for scenes containing an object edge. A pinhole aperture, as shown in Figure 1 (c), is uninfluenced by this abrupt change in depth as it is out of the view of the sensor pixel; this produces an identical distribution of measurements as Figure 1 (a), noisy but correctly centered. A wide aperture setup, however, would integrate light paths of significantly differing length (Figure 1 (d)). This still tightens the expected distribution of measurements, as the increased photon count still reduces the effects of noise and photon shot. Unfortunately, this now tight distribution of measurements is centered somewhere between the true background depth and erroneous foreground depth. If we were not aware of the scene discontinuity, and instead just operating with a set of measurements returned by this sensor pixel, it would look like a confident depth measurement when in fact it is a flying pixel (FP). This demonstrates the other side of the tradeoff, that FP susceptibility is expected to increase as aperture size increases, and vice versa.

As emphasized in the main text, a global aperture setup allows us only to sample spatially homogenous configurations of SNR/FP tradeoff. The system overall can be tuned to prioritize high SNR or high robustness to FPs, but any individual sensor pixel will be just as susceptible to both as its neighbors. Mask-ToF, with its learned microlens mask pattern, can try to optimize for both. With spatially varying susceptibility to noise and flying pixels, pixel neighborhoods can aggregate information to both reduce noise and detect/rectify flying pixels. In the global aperture case we are effectively stacking multiple instances of Figure 1 (d), which, given their colocation, see the same scene discontinuity. This does not provide new information with which we can rectify the measured depth. In the microlens masking case, we can instead imagine pixels neighborhoods containing instances of both Figure 1 (c) and Figure 1 (d). Here the disparity between these pixels' measurements, given that we expect they are observing roughly similar scene content, can both indicate that a flying pixel is present and provide cues to rectify them (e.g. "Trust the measurements of pixels with small aperture masks more since they are less susceptible to FPs.").

## 2. Mask Analysis

In this section, we describe the hand-crafted mask patterns that we compare to and that compose the set of initial iterates for Mask-ToF. We dive into the intuition behind them, and their effects on the final learned mask after training. As we will see, the recursive nature of the Mask-ToF training process means the initial set of masks plays a significant role in the final performance of our method. We additionally explore the importance of added noise in mask evolution, demonstrating pinhole mask regression in its absence.

### 2.1. Spatially Uniform Masks

Spatially uniform masks are the lowest complexity masks, where each sensor pixel receives a roughly identical distribution of light rays. These include the circular aperture masks, as shown in Figure 2, which are generated as Gaussian kernels with variance proportional to the desired diameter. The *Ones* mask is the open-aperture extension of these masks, averaging all views from the input light field data, while the *Diameter 1* mask is its foil, using only the chief rays for reconstruction. For this reason *Diameter 1* is also sometimes referred to as the *Pinhole Mask*. As the diameter of these masks increase, so typically does the reconstruction SNR, as the added Gaussian noise from multiple views is averaged out. This, however, also leads to more flying pixels (FPs), as the valid integration range over the camera aperture is increased and there is a wider range of angles from which an object boundary is able to interfere with a sensor pixel's measurements. Testing these masks

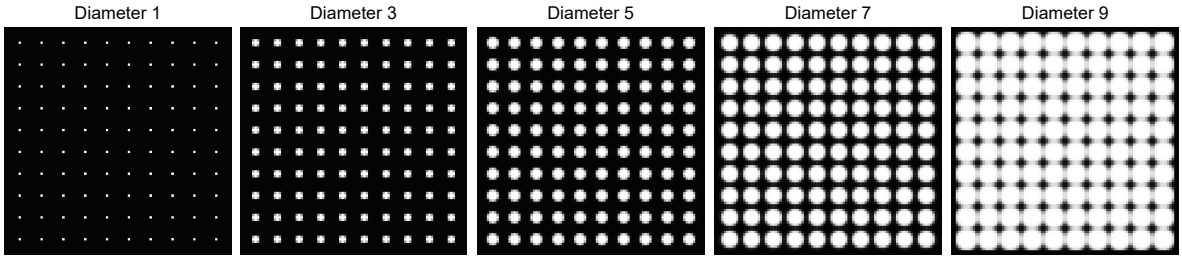


Figure 2: Example circular aperture masks for diameters 1-9 ($90\text{px} \times 90\text{px}$ un-binarized regions shown).

thus gives us an effective way to observe and quantify the SNR-FP tradeoff.

We also consider uniformly random initialized masks, such as white Gaussian noise or a Bernoulli sampling of on and off bits, as spatially uniform. Although sensor pixels may receive different distributions of light, the local mask-distributions are un-oriented, providing little geometric information. During experimentation it was found that many forms of random initializations behave identically during training. As it is by definition binarized, and so requires no augmentation to physically manufacture, the *Bernoulli* mask was chosen as a representative candidate for these uniformly random masks.

### 2.2. Spatially Multiplexed Masks

Spatially multiplexed masks are those in which a sensor pixel receives either directionally oriented light, such as that passing through a vertical or horizontal slit, or a significantly different amount of light as compared to its neighbors. Intuitively in a raw reconstruction pipeline, where each pixel of depth is estimated without consideration for the local pixel neighborhood, this leads to unavoidable noise as each pixel blends a different proportion of light from the scene. Given a refinement network $R$, however, this erstwhile *noise* can instead be used as encoded information to estimate local geometry and disambiguate flying pixels.

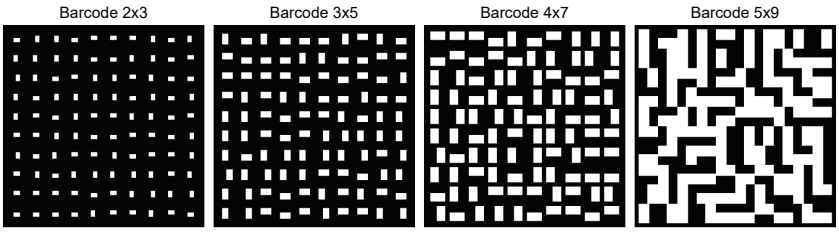


Figure 3: Example *Barcode Masks* ($90\text{px} \times 90\text{px}$ regions shown).

One type of multiplexed mask candidate is the so-called *Barcode* mask, named after its resemblance to a commercial bar or QR code. These masks, shown in Figure 3, are made up of randomly oriented horizontal and vertical apertures, with the $\boldsymbol{X}\times\boldsymbol{Y}$ in *Barcode* $\boldsymbol{X}\times\boldsymbol{Y}$ referring to the dimensions of the rectangular aperture. Each aperture is uniformly randomly set to be in one of four possible orientations: North (N), South (S), East (E), and West (W). As an example, the top row of *Barcode 4x7* in Figure 3 is the sequence [N,N,N,S,W,S,N,E,E,E]. This type of initial mask encourages the network $R$ to learn the relation between mask orientation and flying pixel formation, for example if neighboring pixels with N and E directed apertures return vastly different depths, this implies the existence of either a vertical or horizontal depth discontinuity. Surrounding pixels can be then be used to identify whether N or E is a more reliable measurement, and thus use their combined information to disambiguate the true depth at each pixel.

A contrasting set of multiplexed mask candidates are the *Gaussian Circle* masks. In these masks, each microlens aperture is a circular $9 \times 9$ Gaussian kernel, shifted to the range [0,1], whose standard deviation is determined by a random draw from a Gaussian distribution with a chosen mean $\mu_{GC}$ and standard deviation $\sigma_{GC}$. Several example masks for varying $\mu_{GC}$ and

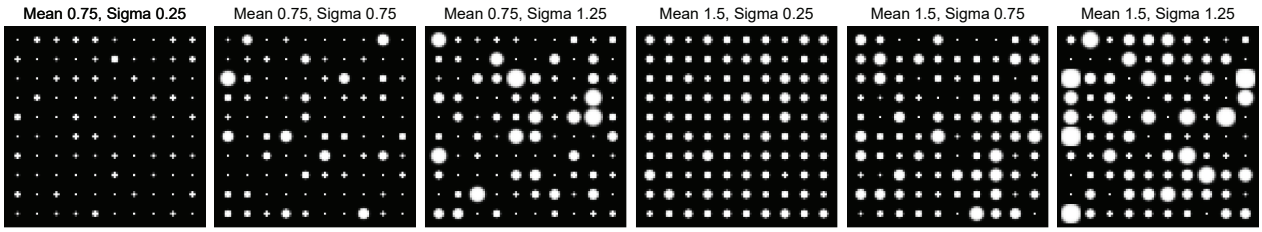


Figure 4: Example *Gaussian Circles* masks for several values of mean and standard deviation ($90\text{px} \times 90\text{px}$ regions shown).

$\sigma_{GC}$ are visualized in in Figure 4. This process allows us to control average light throughput for these masks by tuning $\mu_{GC}$, and the level of multiplexing by modifying $\sigma_{GC}$. Here, as each initial microlens mask is essentially a collection of circular apertures with random radii, the idea is that we encourage $R$ to learn a relation between light throughput and flying pixel formation. Neighboring pixels thus have varying susceptibility to noise and flying pixels, and by aggregating their combined information $R$ can both estimate non-flying depth measurements and suppress noise.

### 2.3. Influence of Initialization on Mask Evolution

The Mask-ToF end-to-end training process contains a feedback loop: a change in the mask structure requires an update in $R$ to process the reconstructed depth outputs, and an update in $R$ alters the propagated loss gradient, which itself changes the mask structure. This, when combined with the fact that we do not update the mask in the initial noisy training epochs, means that the initial mask iterate plays a great role in the evolution of the mask design during training. Figure 5 illustrates some key observations in mask evolution for a set of spatially uniform and multiplexed initializations.

One of the key observations is that while the *Ones* and *Bernoulli* masks appear to be very different initializations, they both converge on nearly the same uniformly circular mask. Given this, we posit that the driving force in mask evolution is whatever the largest source of error is at a particular epoch. For the initial *Ones* mask there is little random noise, and instead the majority of this error comes from incorrectly predicted object boundaries (flying pixels). As these FPs heavily stem from wide-angle views, we see that the method learns to remove the edges of each microlens aperture, whittling down the mask as much as it can without incurring heavy penalties from noise passthrough. In the evolution of the *Bernoulli* mask we observe that it initially fills in the centers of each microlens mask, as a significant source of error at initialization is from random noise. Once this infill is achieved, around epoch 30, the primary source of error becomes once again these wide-angle views and edge discontinuities. Thus the mask evolution follows the same trajectory as the latter part of *Ones*.

We note next that in the interest of training noise suppression, we must limit the mask's learning rate and overall network training length. Otherwise by the time the refinement network $R$ begins to learn high-level features, any existing mask structure would already be erased. This results in two phenomena. Firstly, that there is a fundamental limit to how far a mask can evolve from its initialization. We observe that *Diameter 5* appears to be a stable local minimum, not changing over 500 epochs, implying that it is, for the given noise level, a good tradeoff between SNR and flying pixel suppression. But while the *Ones* mask appears to decrease in aperture size over training, and *Diameter 1* appears to increase, due to this fundamental limit neither quite manage to reach the *Diameter 5* local minimum. The second phenomenon is that local minima are particularly difficult to escape. Spatially uniform patterns do not appear to evolve into more complex, yet lower loss (see Table 14), spatially multiplexed designs. We theorize that if $R$ does not learn to use spatial multiplexing during its initial epochs, it suppresses the development of non-uniform features during later stages of training. What is of interest, however, is that in the example of the *Gaussian Circles* initialization, the mask not only retains its spatial multiplexing, but also promotes vertically oriented microlens masks. We theorize that the network may be learning to take advantage of signal statistics during reconstruction, biasing aperture shape by the distribution of observed horizontal and vertical edges in the training data. To choose our final mask for Mask-ToF, we compare a wide range of handcrafted masks and several hundred

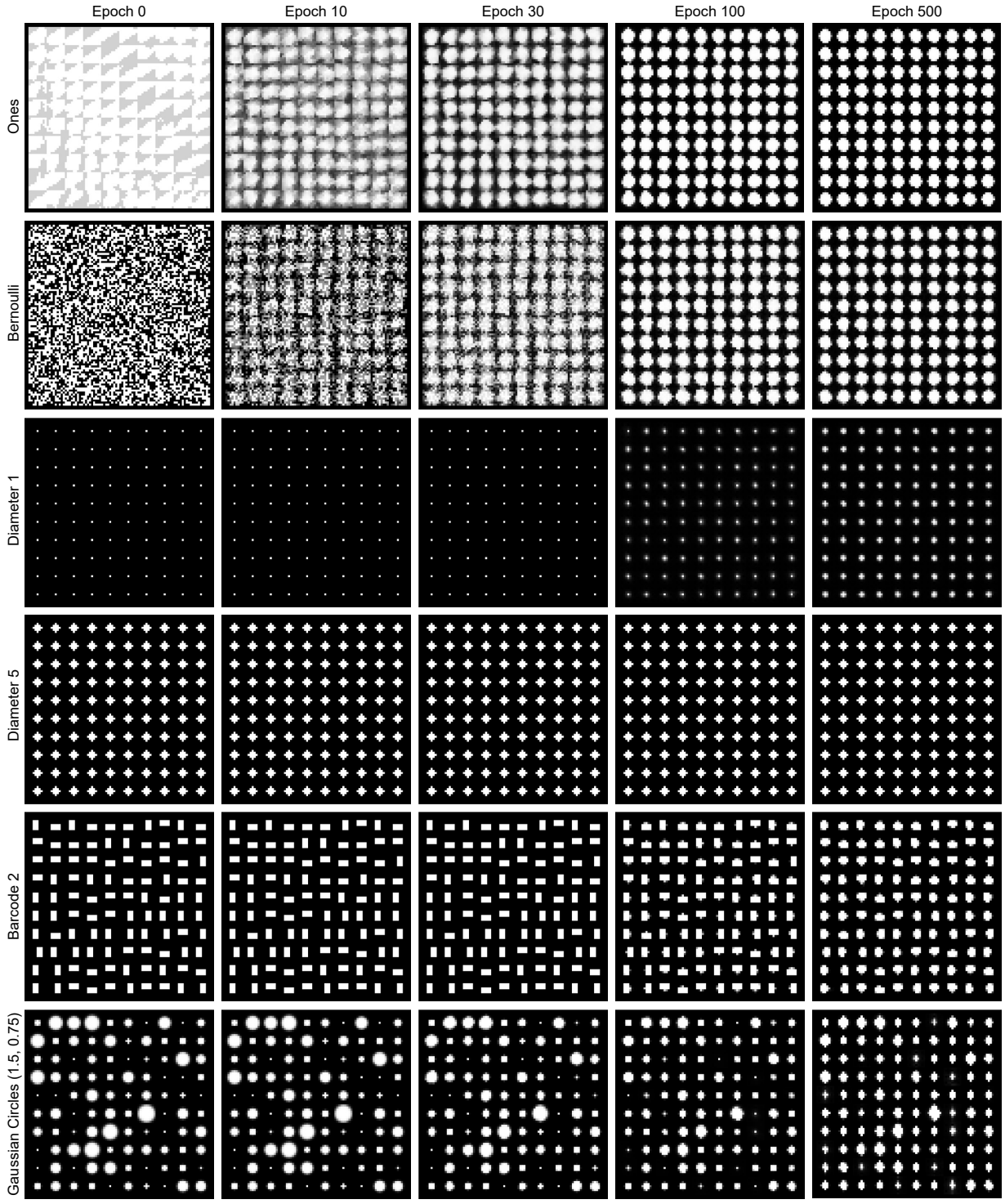


Figure 5: Visualization of mask evolution during training for sample spatially uniform and multiplexed initial mask patterns (90px $\times$ 90px regions shown). Noise, network, and training parameters used are the same as those in the final Mask-ToF design in the main text.

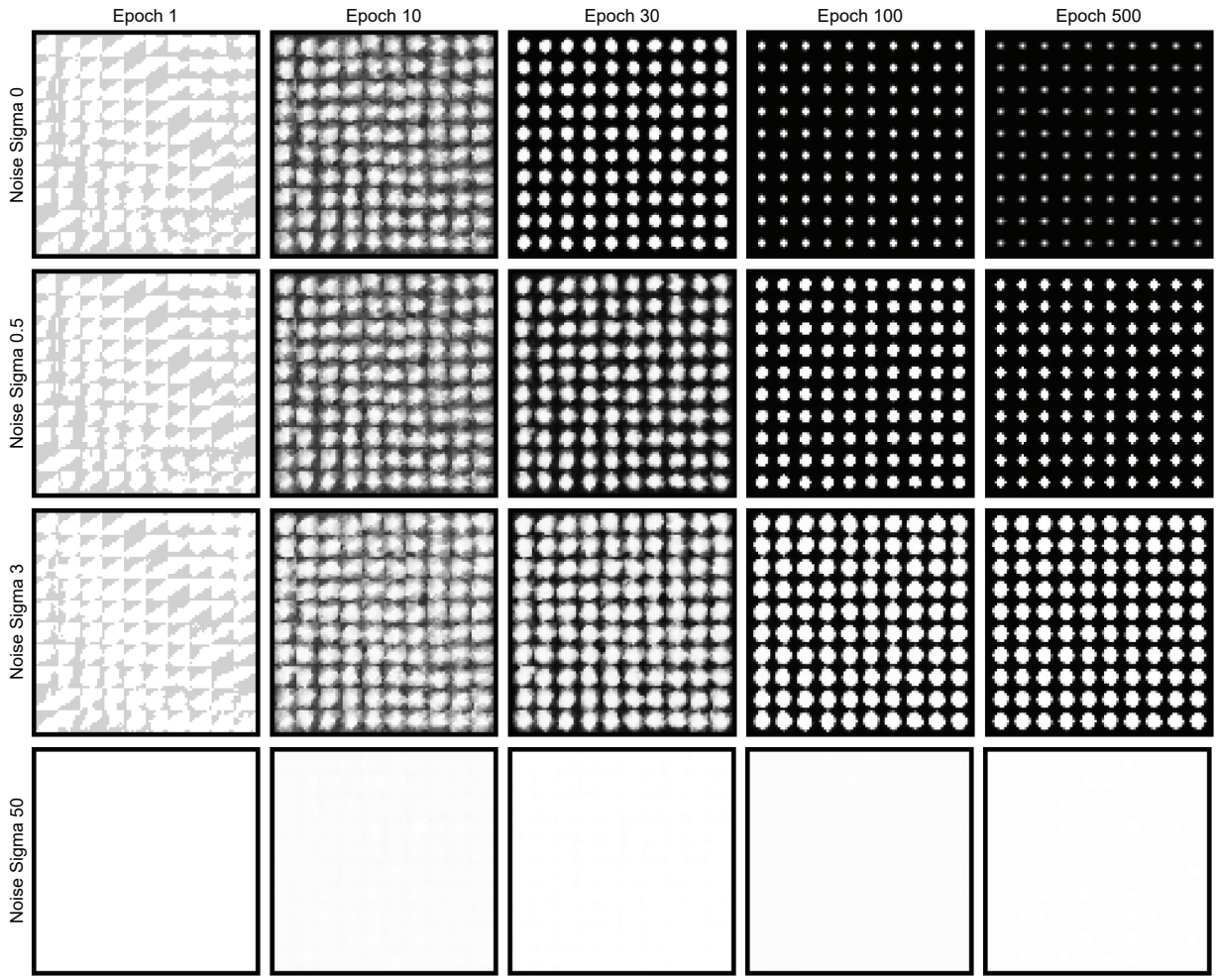


Figure 6: Evolution of masks for varying standard deviations of added Gaussian noise, all with zero mean and ($90\text{px} \times 90\text{px}$ regions shown). Epoch 0 not displayed as the initial mask is *Ones* for all tests.

initializations of random multiplexed initial iterates (annealing over values of $\mu_{GC}$ and $\sigma_{GC}$).

### 2.4. Influence of Noise on Mask Evolution

Given a synthetic dataset, noise is essential to the training process. Without simulated system noise, the only source of error is from flying pixels, and so the *Diameter 1* pinhole mask results in zero error. In the top row of Figure 6, we see that with no added noise, the learned mask rapidly converges to this pinhole pattern as the *correct* optimum which minimizes flying pixels. In the bottom row we see the complementary sanity check, where the noise added is so large it overpowers any real signal, and the mask maximizes light throughput by remaining at the initial open aperture pattern. The third row visualizes the evolution of the *Ones* mask for our chosen noise level, where we see it neither devolves to the pinhole or remains at the *Ones* pattern, instead leading to a mask with approximately 50% light throughput. At this level it approximates real levels of time-of-flight (ToF) imaging noise.

## 3. Prototype Fabrication

In this section, we provide additional details on the experimental prototype we designed to validate Mask-ToF.

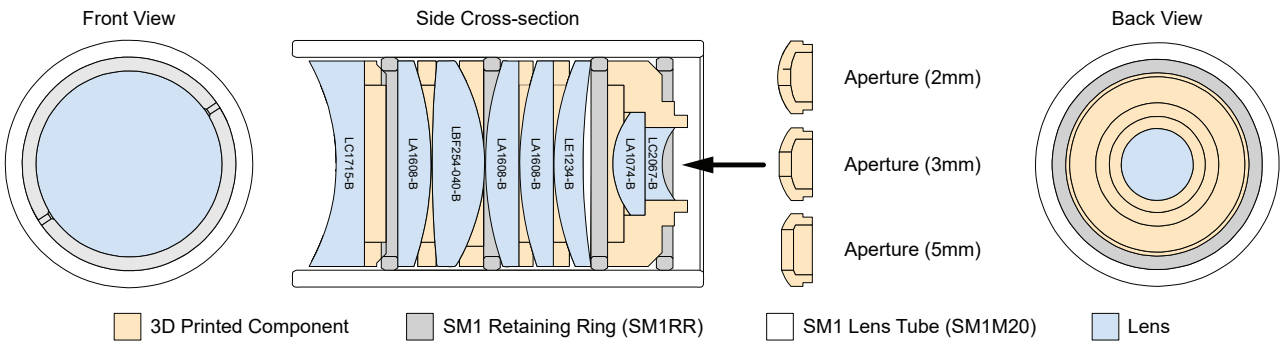


Figure 7: Visualization of half of the optical relay assembly with Thorlabs part-numbers; the two halves are symmetric and use the same parts. Custom spacers were manufactured via SLA 3D printing. Three different diameters of aperture (2mm, 3mm, and 5mm) were created to be slotted into the back of the assembly, the 5mm diameter aperture was chosen for all main and supplemental experiments.

As the CMOS ToF sensor of our Lucid Vision Helios Flex camera is covered by a glass window with an air gap, it is impractical to place masks directly onto the sensor plane for proof-of-concept experiments without risking permanent damage to the sensor. In iterations closer to a commercial product, such a mask could be directly integrated in the sensor fabrication process during camera manufacturing. For our prototype we opt for an optical relay approach to virtually place the mask on the sensor's image plane. An ideal Keplerian telescope design should cover the full field of view of the ToF camera ($\pm 33.7°$) and match its exit pupil with the entrance pupil of the ToF lens. The front group of the telescope should also form a high-quality image of the scene on the intermediate image plane for masking. We note that it is impossible to meet all these requirements with singlets or achromats, and we would need specifically designed lens groups to correct for all optical aberrations. For our prototype we opt to design a close approximation to this ideal system with off-the-shelf components, leading to a complex 16 lens assembly as shown in Figure 7. We will release the Zemax design files and SolidWorks assembly files for the employed optical system.

We design the 1:1 telescope by combining two identical wide-angle eyepiece-type lens groups, each with 8 intermediate lenses, with the back focal plane of the first coincident on the front focal plane of the second. The entrance pupil of the first group is designated to be in the very front of the system, such that the exit pupil of the whole system is after the second group to allow all collected light to go into the entrance pupil of the ToF lens. The telescope is completely symmetric about the mask plane, so off-axis aberrations are canceled out, resulting in a high-quality image on the ToF camera. The magnification ratio from the mask plane to the sensor plane is determined by the ratio (2.298:1) between the focal length of the rear group of the telescope (13.788 mm) and the focal length of the ToF lens (6 mm).298. This means we scale up the sensor pixel size ($10\mu$m) by a factor of 2.298 for the mask pixel size (now $22.98\mu$m). This additional scale factor reduces requirements on the mask fabrication accuracy. To account for residual distortions, we pre-warp masks before fabrication, as outlined in the next section. Nonetheless, at the edges of the image plane optical aberrations and hindered light propagation from the enclosing lens tube leads to loss in SNR and unwanted distortions. Thus, for experimental analysis we crop a central circular region with a diameter of approximately 300px where mask effects on light propagation dominate aberrative light collection.

We fabricate the masks via photolithography. A master mask is first laser direct written on a 5-inch soda lime substrate with a tabletop maskless aligner system (Heidelberg $\mu$PG501). We use a 4-inch fused silica wafer ($0.5mm$ thick) as the substrate for the mask. A $200nm$ thick Chromium (Cr) film is deposited by sputtering on one side of the wafer, and a layer of $0.6\mu m$ thick photoresist AZ1505 is then spin-coated on top of the Cr film. We place the wafer under the master mask on a contact aligner EVG 6200$\infty$ for UV exposure. The wafer is then developed in AZ726 for 20s to form the mask pattern on the

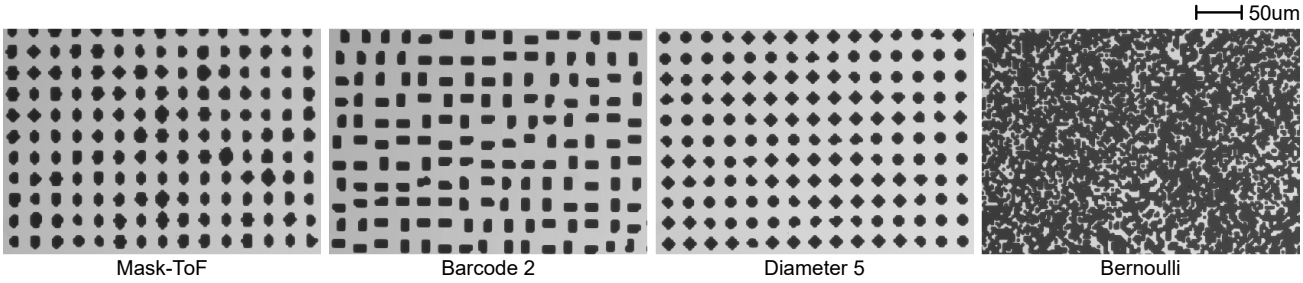


Figure 8: We fabricate four mask designs: the final optimized *Mask-ToF* mask, *Barcode 2*, *Diameter 5*, and a random *Bernoulli* pattern. We capture images of the fabricated amplitude masks using a microscope with micrometer resolution. Note that these masks are pre-warped, hence after interpolation and binarization individual microlens apertures may deviate in shape from their original mask patterns.

photoresist. We etch the Cr under the open areas in the photoresist via a Cr etchant for $2.5min$. The residual photoresist is then removed with acetone. The wafer is diced into $17mm \times 20mm$ samples to produce the final masks. Microscope images of these final fabricated masks are shown in Figure 8.

## 4. Artifacts in Experimental Data

In this section, we discuss some of the ways in which acquired experimental data deviates from simulation, and how these deviations can be addressed. We emphasize that most, if not all, of these unwanted effects are byproducts by the optical relay system, not inevitable issues for a masked ToF system. We expect this section to be superfluous for a Mask-ToF system with the mask implemented directly on the ToF camera sensor.

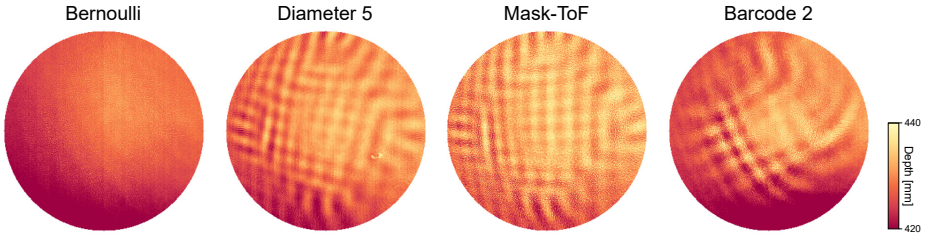


Figure 9: (a) Barrel warp applied by first half of the optical relay. (b) Pincushion warp applied by second half of the optical relay. (c) Example unmodified mask. (d) Pre-warped mask with added alignment border.

### 4.1. Pre-Warping

Considering the large field of view and pupil matching requirement for the optical relay system, there is an inevitable barrel warp on the intermediate image plane, as shown in Figure 10 (a). The latter half of the optical relay produces a complementary pincushion warp, Figure 10 (b), which rectifies the shape of the resultant image on the sensor plane. However, as the mask sits at the intermediate image plane, we must *pre-warp* it prior to manufacturing to account for this distortion and insure it is applied correctly. We note that such compensation would *not be required for a mask that is fabricated on the sensor plane*. We are able to estimate this distortion in simulation, and augment the mask via backward warping as shown in Figure 10 (c) and (d). We additionally apply a border of open aperture pixels to the edges of this pre-warped mask to assist with the alignment process. By observing the live amplitude output of the ToF camera we can adjust the X and Y position of the mask, via the translation stage, until these edges disappear from view.

### 4.2. De-Warping

Other unavoidable artifacts in the experimental prototype are interference patterns from light diffraction through the microscale mask structure, illustrated in Figure 10. Without a relay optic, with masks fabricated on the sensor plane, these diffraction patterns likely would not occur as there is no free space behind the mask for the light to propagate in. Something of note is that the spatially random *Bernoulli* mask exhibits few to none of these artifacts, as the overall effect of light diffraction is destructively averaged out by its random distribution throughout the mask.

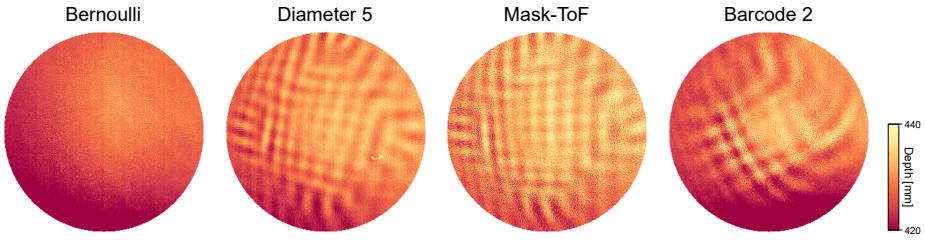


Figure 10: Diffraction patterns observed in unprocessed experimental measurements with fabricated masks; depth target is a flat wall. Note that spatially random *Bernoulli* mask does not exhibit constructive interference patterns.

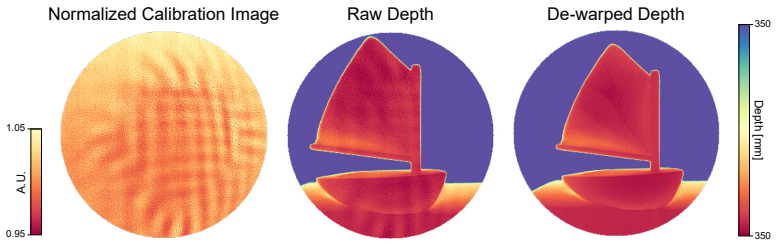


Figure 11: Example normalized calibration pattern, raw measured depth map, and de-warped depth map.

A simple but effective method for removing these artifacts is de-warping with reference calibration data. For each mask pattern and test object, we additionally collect calibration images of a flat wall at the depth of the object plane and background plane. These calibration measurements are required as the diffraction pattern is depth dependent. To produce evaluation data, the collected raw depth measurements are first split into object and background points based on their relative depth. These are then divided by the normalized depth values of the corresponding points in the respective calibration images. The result of this process is shown in Figure 11. For objects or scenes that span a large range in depth, we suggest to acquire several calibration images (more than two) at known depths and generate a 3D interpolation matrix to map raw depth points $d_{\mathrm{RAW}}(x, y, z) \rightarrow d_{\mathrm{DE-WARPED}}(x, y, z)$. We note that range-based calibration is typical in ToF imaging.

### 4.3. Other Sources of Experimental Error

The optical relay, translation stage, and mounting equipment used to keep these components in place unfortunately block a considerable amount of light from the ToF camera's LED array. The bottom two of the four near-infrared illumination LEDs are completely blocked as they cast strong shadows on the scene, which create reflectance-dependent depth reconstruction artifacts that cannot be calibrated out post-capture. The top two LEDs are thus the only source of illumination, and lead to lower-than-expect light throughput even before masking. Exposure time is limited by the camera software, and thus the resulting scans have elevated background noise as seen in Figures 17 through 25.

## 5. Network Details

In this section, we provide an overview of our refinement network $R$, its implementation, and its role in mask evolution.

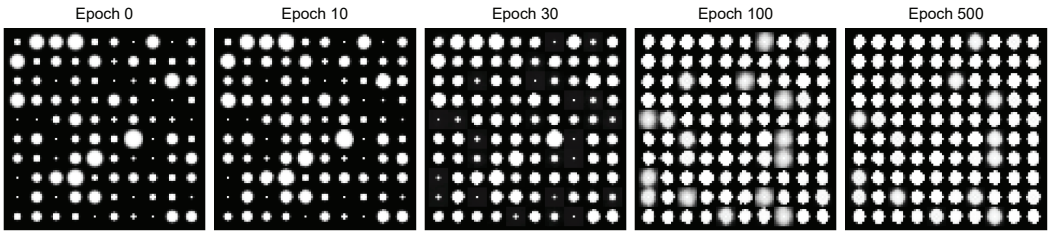


Figure 12: Without the refinement network $R$, training initialized with the spatially multiplexed *Gaussian Circles* pattern ($\mu_{GC} = 1.5, \sigma_{GC} = 0.75$) converges to a spatially uniform mask structure. Compare this to the evolution of the same mask in the bottom row of Figure 5. ($90\text{px} \times 90\text{px}$ regions shown).

$R$, with its architecture outlined in detail in Table 13, is a residual encoder-decoder model whose primary role during training is to aggregate and utilize neighborhood information to refine depth estimates. This refinement network is essential to both the mask evolution process and final depth reconstruction fidelity, as without it neighboring pixels cannot share information and so high-level multiplexed mask patterns cannot arise. This is demonstrated in Figure 12, wherein without $R$ an initially spatially multiplexed mask design devolves into a local minimum spatially uniform pattern.

Inspired by the hourglass architecture from [2], the goal of this design is to quickly learn high-level feature correspondences to produce a depth residual term $\hat{D}^R$. When added to the unrefined depth estimate $\hat{D}$, this term serves to correct localized outliers such as spatially multiplexed FPs, and reduce fluctuations caused my noise in otherwise smooth regions. Thanks to the abundance of skip layers within the network, as well as this residual connection from input to output, $R$ can quickly learn salient depth and mask features without being burdened by the task of having to learn to reconstruct depth from scratch. By directly processing depth can also rely on calibration procedures implemented by the original camera manufacturers [1]. This allows us to directly feed real ToF camera data into $R$ without the need for retraining or learning calibration offsets from data.

We implement our running mask variable as a $9\times9\times2\times80\times80$ array, where $80\times80$ is the mask patch size, and for each mask pixel there are $9\times9$ views from the light field data. Rather than explicitly representing the mask as a floating-point array and clamping it to the range 0-1 (0 meaning we block all light, and 1 meaning we pass light), we add an extra dimension: dimension 2. Before applying the mask to our light field data, we take the softmax of this underlying variable along dimension 2, and regard the second $9\times9\times80\times80$ entry of the resultant array as the explicit mask representation. This means our underlying model is a probability distribution of how likely each mask pixel is to be on or off, and we avoid gradient issues caused by tensor clamping. This allows for a natural process to binarize the mask in preparation for fabrication. We can add a temperature variable $\xi$ which scales the output of this softmax function, and my increasing $\xi$ during training while reducing learning rate we push this underlying variable to a binarized solution. Practically, however, we find that this is not necessary as during training the mask patterns naturally converge to binary solutions, as can be seen in Figure 5.

## 6. Additional Synthetic Results

In this section, we present additional qualitative and quantitative results for representative hand-crafted mask designs as well as our optimized mask pattern. Figures 15 and 16 display depth estimates with zoomed in views on regions of interest (ROIs). It is important to note that flying pixels are depth and focus dependent artifacts, and so may not be equally present over the range of a scene. For example, the building on the left of the *Tower* example is at infinity and sharply rendered in all reconstructions. The foreground information on the right of this example, however, exhibits severe flying pixel artifacts. The

Figure 13: Refinement network $R$ architecture, split into *Hourglass 1* and *Hourglass 2*. In the table *conv(a,b,c,d)* represents a 2D convolutional layer with a kernel size $a$, stride $b$, dilation $c$, and padding $d$, and *convT* is transposed convolution. BN denotes batch normalization, *ReLU* is the rectified linear unit, and *LReLU* is leaky ReLU ($\alpha = 0.2$). The notation + represents a stacking of operations (i.e. *conv(3,3,3,2) + BN + ReLU* would be a 2D convolution followed by batchnorm and ReLU).

| Hourglass 1 **(H1)** | | | Hourglass 2 **(H2)** | | |
|---|---|---|---|---|---|
| Layer Name | Type | Channels | Layer Name | Type | Channels |
| input1 | mask | 1 | conv1b | conv(3,2,1,1) + BN + ReLU | 64 |
| mask_conv1 | conv(3,3,3,2) + BN + ReLU | 16 | | concat(deconv2a) | 128 |
| mask_conv2 | conv(3,3,3,2) + BN + ReLU | 1 | | conv(3,1,1,1) + BN + ReLU | 64 |
| input2 | depth | 1 | conv2b | conv(3,2,1,1) + BN + ReLU | 128 |
| concat1 | concat(mask_conv2, Input2) | 2 | | concat(deconv3a) | 256 |
| conv1 | conv(3,1,1,0) + BN + LReLU | 16 | | conv(3,1,1,1) + BN + ReLU | 128 |
| conv_start | conv(1,1,1,0) + BN + ReLU | 32 | conv3b | conv(3,2,1,1) + BN + ReLU | 256 |
| conv1a | conv(3,2,1,1) + BN + ReLU | 64 | | concat(deconv4a) | 512 |
| conv2a | conv(3,2,1,1) + BN + ReLU | 128 | | conv(3,1,1,1) + BN + ReLU | 256 |
| conv3a | conv(3,2,2,2) + BN + ReLU | 256 | conv4b | conv(3,2,1,1) + BN + ReLU | 512 |
| conv4a | conv(3,2,2,2) + BN + ReLU | 512 | | concat(conv4a) | 1024 |
| deconv4a | convT(4,2,1,1) + BN + ReLU | 256 | | conv(3,1,1,1) + BN + ReLU | 512 |
| | concat(conv3a) | 512 | deconv4b | convT(4,2,1,1) + BN + ReLU | 256 |
| | conv(3,1,1,1) + BN + ReLU | 256 | | concat(conv3b) | 512 |
| deconv3a | convT(4,2,1,1) + BN +LReLU | 128 | | conv(3,1,1,1) + BN + ReLU | 256 |
| | concat(conv2a) | 256 | deconv3b | convT(4,2,1,1) + BN + ReLU | 128 |
| | conv(3,1,1,1) + BN + ReLU | 128 | | concat(conv2b) | 256 |
| deconv2a | convT(4,2,1,1) + BN + ReLU | 64 | | conv(3,1,1,1) + BN + ReLU | 128 |
| | concat(conv1a) | 128 | deconv2b | convT(4,2,1,1) + BN + ReLU | 64 |
| | conv(3,1,1,1) + BN + ReLU | 64 | | concat(conv1b) | 128 |
| deconv1a | convT(4,2,1,1) + BN + ReLU | 32 | | conv(3,1,1,1) + BN + ReLU | 64 |
| | concat(conv_start) | 64 | deconv1b | convT(4,2,1,1) + BN + ReLU | 32 |
| | conv(3,1,1,1) + BN + ReLU | 32 | | concat(deconv1a) | 64 |
| | | | | conv(3,1,1,1) + BN + ReLU | 32 |
| | | | final_conv | conv(3,1,1,1) + BN + LReLU | 1 |
| | | | output | sum(input2, final_conv) + ReLU | 1 |

Figure 14: Additional synthetic results for spatially uniform and multiplexed masks. Here $\text{GC}_{a,b}$ refers to the *Gaussian Circles* mask with $\mu_{GC} = a$ and $\sigma_{GC} = b$.In the *Finetuning Only* results, the initial mask was not updated, and only the refinement network $R$ was finetuned. For *Joint Training*, the initial mask was unlocked after epoch 70, as in the main text, and allowed to jointly update with $R$.

| Finetuning Only (No Mask Update) | | | | |
|---|---|---|---|---|
| Mask | RMSE | MAE | Thresh 3mm | Thresh 15mm |
| Diam. 1 | 9.412/8.293 | 5.203/4.576 | 46.31/46.55 | 6.647/4.345 |
| Diam. 3 | 7.284/7.667 | 3.324/2.904 | 26.27/24.19 | 3.652/2.374 |
| Diam. 5 | 6.512/8.732 | 1.718/1.753 | 7.377/6.552 | 1.585/1.582 |
| Diam. 7 | 6.734/9.148 | **1.517**/1.624 | **5.275**/5.526 | 1.745/1.847 |
| Diam. 9 | 8.751/11.77 | 2.497/2.872 | 11.19/13.90 | 2.896/3.292 |
| Ones | 9.227/12.58 | 2.470/2.814 | 9.712/10.45 | 3.118/3.558 |
| Bernoulli | 9.192/12.50 | 2.388/2.707 | 8.894/9.595 | 3.090/3.557 |
| Barcode 2 | 6.821/9.060 | 1.518/**1.580** | 5.461/**5.442** | 1.632/1.676 |
| Barcode 3 | 8.117/10.52 | 1.850/1.913 | 6.355/6.647 | 2.173/2.172 |
| Barcode 4 | 9.312/12.01 | 2.307/2.457 | 8.554/9.424 | 2.763/2.938 |
| $\text{GC}_{0.75,0.75}$ | 6.028/**6.395** | 2.695/2.458 | 22.00/20.66 | 2.053/1.485 |
| $\text{GC}_{1.50,0.75}$ | **5.843**/7.269 | 1.902/1.831 | 10.63/9.448 | **1.579**/**1.428** |

| Joint Training (With Mask Update) | | | | |
|---|---|---|---|---|
| Initial Mask | RMSE | MAE | Thresh 3mm | Thresh 15mm |
| Diameter 1 | 5.455/**6.941** | 1.850/1.727 | 10.83/8.521 | 1.478/1.242 |
| Diameter 3 | 5.394/7.113 | 1.757/1.695 | 10.12/8.011 | 1.402/1.322 |
| Diameter 5 | 6.180/8.668 | 1.358/1.462 | **4.912**/4.956 | 1.501/1.596 |
| Diameter 7 | 6.759/9.136 | 1.529/1.648 | 5.403/5.654 | 1.812/1.956 |
| Diameter 9 | 7.882/10.56 | 2.208/2.382 | 9.134/9.649 | 2.393/2.553 |
| Ones | 7.553/9.998 | 1.785/1.913 | 6.262/6.733 | 2.135/2.261 |
| Bernoulli | 7.561/10.00 | 1.765/1.898 | 6.288/6.679 | 2.123/2.297 |
| Barcode 2 | 6.246/8.585 | 1.398/1.460 | 5.054/4.916 | 1.456/1.488 |
| barcode3 | 7.581/9.954 | 1.870/1.926 | 7.314/7.261 | 2.026/2.027 |
| barcode4 | 8.531/10.82 | 2.257/2.300 | 8.786/8.839 | 2.688/2.662 |
| $\text{GC}_{0.75,0.75}$ | 5.314/7.211 | 1.673/1.618 | 8.782/7.133 | 1.371/1.260 |
| $\text{GC}_{1.50,0.75}$ | **5.166**/7.115 | **1.281**/**1.278** | 5.052/**4.397** | **1.178**/**1.120** |

ROIs are selected to highlight these problematic edge areas and Mask-ToF's ability to reconstruct the original features.

Table 14 compiles quantitative results in a number of metrics for a wide range of initial mask iterates. The left sub-table reviews results given only fine-tuning of the refinement network $R$, leaving the initial mask unchanged throughout training, and the right table demonstrates results given joint learning of the mask and $R$. We see that while $GC_{1.50,0.75}$, the initial mask iterate used in Mask-ToF, shows lackluster performance with fine-tuning only, it shows stellar performance when allowed to jointly train with $R$. It goes from scoring poorly in MAE and the 3mm threshold metrics, to top score in nearly all categories. This reinforces the value of this joint training and a well-selected initial mask structure. Another surprising finding is the *Diameter 5* mask, which we saw does not evolve into a new pattern (Figure 5), still performs better with joint training. This implies that the feedback loop and loss propagated between $R$ and the mask might serve to regularize and improve training by itself.

## 7. Additional Experimental Results

In this section, we showcase additional experimental results for the fabricated masks introduced in Figure 8. In Figures 17 - 25 we see point cloud projections of reconstructed depth, which help to visualize flying pixel removal for a geometrically diverse set of objects. These range from the flat *Club* target to the round plaster busts and jagged *Plant* examples. These further validate the claims in the main document, as we see a noticeable reduction in flying pixels by Mask-ToF as compared to all hand-crafted mask designs. Also of note is in Figure 25, Mask-ToF is able to reconstruct the raised specular surfaces in the *Cat* example as well as the open aperture measurements. We posit that this is due to its multiplexed design, the interspersed large diameter sub-apertures allowing for sufficient light input to disambiguate local pixel neighborhoods in these specular regions. Conversely, our *Barcode 2* hand-crafted mask design suffers heavily in these regions, as its thin oriented mask structure means there is little to no modulated light that can be observed by the camera for these slanted specular surfaces in out-of-view directions.

| | Ones | Bernoulli | Diameter 5 | Barcode 2 | Mask-ToF |
|---|---|---|---|---|---|
| Flying Pixel Ratio: | 1 | 0.82 | 0.69 | 0.72 | 0.48 |

We quantify flying pixel removal via the same flat target analysis as in the main text, and report flying pixel ratios in Table 7. We find that the *Bernoulli* mask is not very effective at reducing FP counts, *Barcode 2* and *Diameter 5* produce similar FP counts, but Mask-ToF is the clear winner. It provides a 30.5% reduction in FPs over the *Diameter 5* mask and a 33.3% reduction as compared to *Barcode 2*.

## References


[1] A. Kolb, E. Barth, R. Koch, and R. Larsen. Time-of-flight cameras in computer graphics. In *Computer Graphics Forum*, volume 29, pages 141–159. Wiley Online Library, 2010. 10

[2] H. Xu and J. Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020. 10

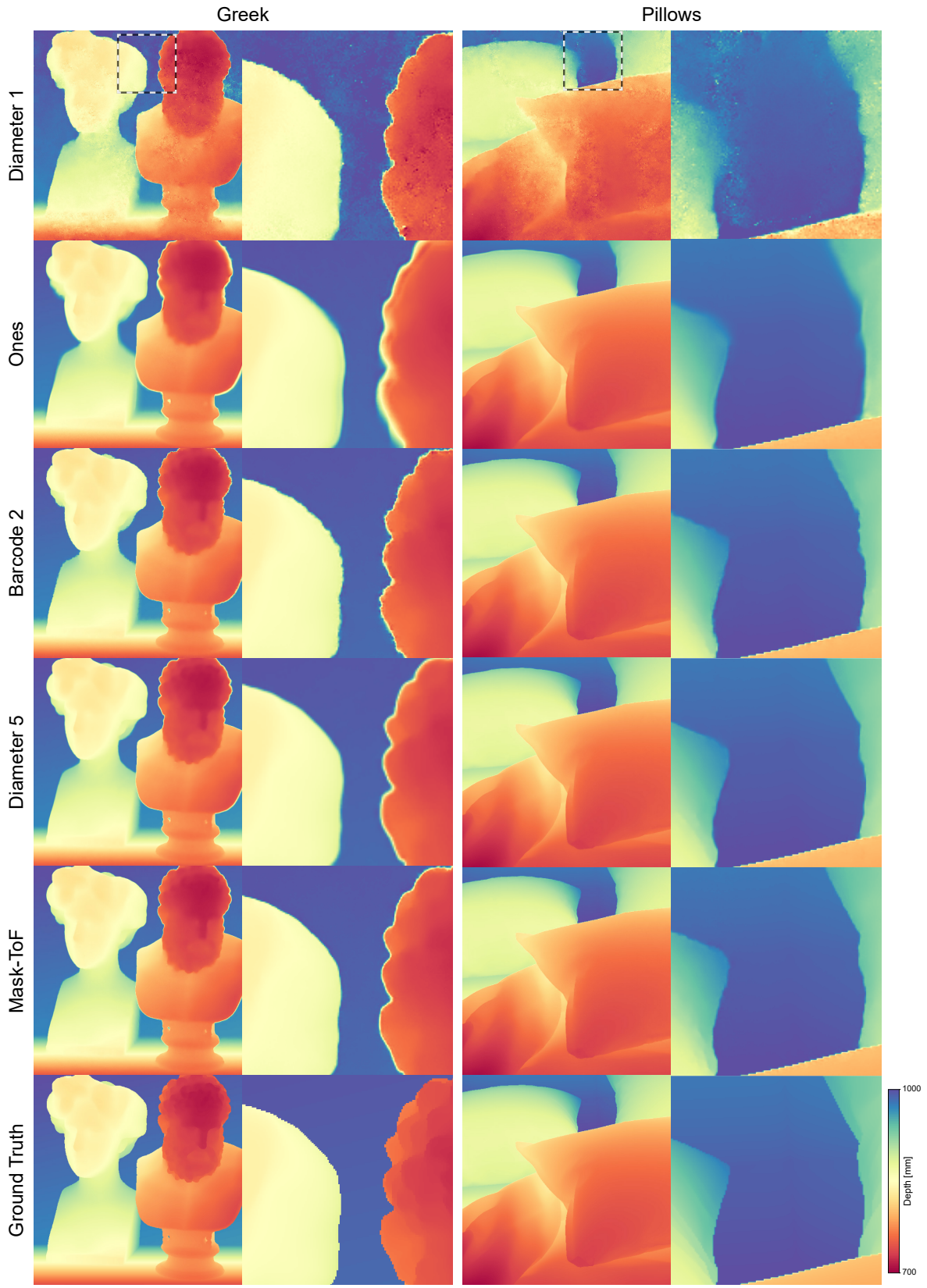


Figure 15: Comparison of depth reconstructions for optimized and naïve mask designs; test samples *Greek* and *Pillows*. Zoomed views shown to highlight artifacts on object boundaries, with ROI borders displayed in top row.

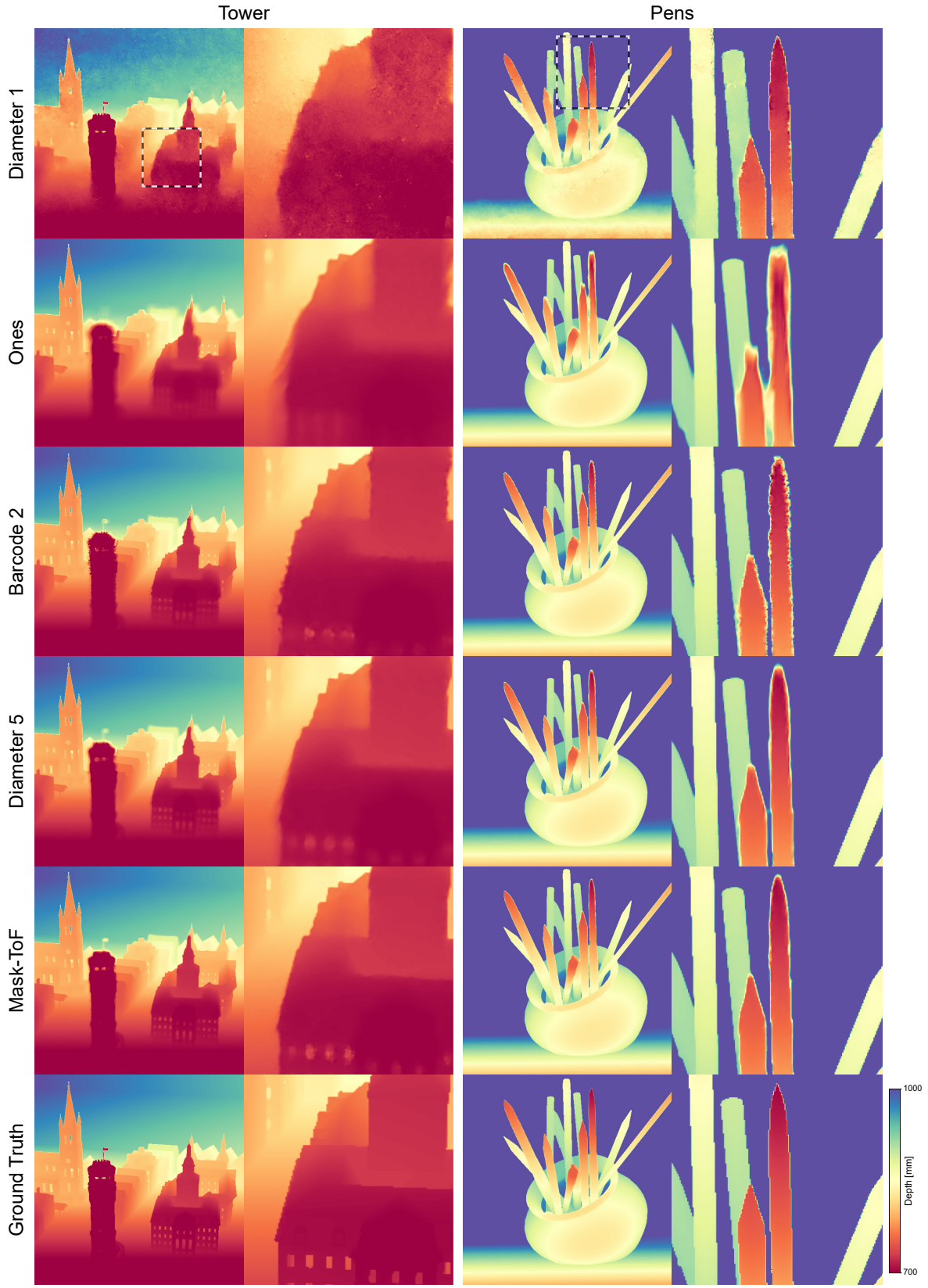


Figure 16: Comparison of depth reconstructions for optimized and naïve mask designs; test samples *Tower* and *Pens*. Zoomed views shown to highlight artifacts on object boundaries, with ROI borders displayed in top row.

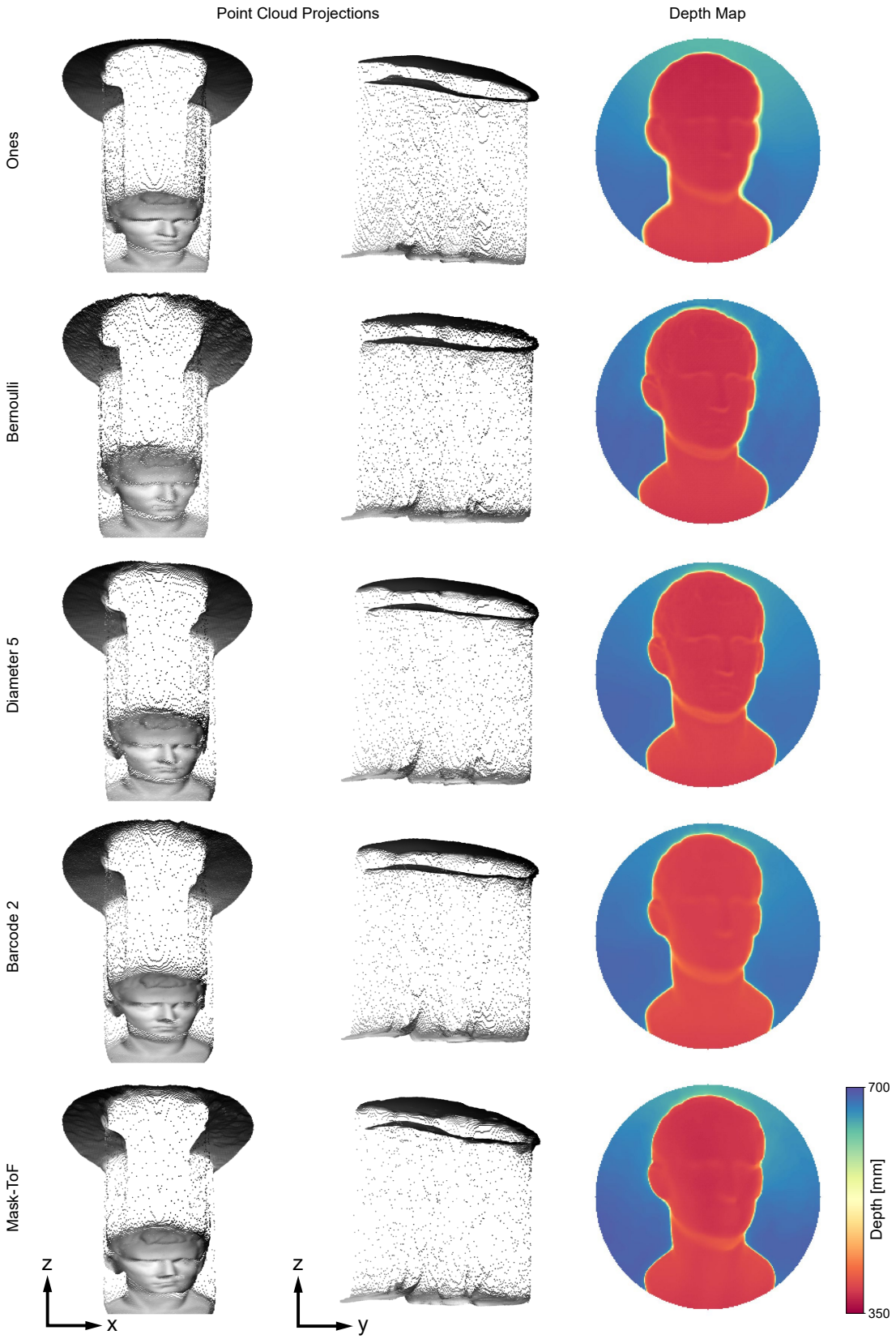


Figure 17: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Agrippa*, a replica plaster bust.

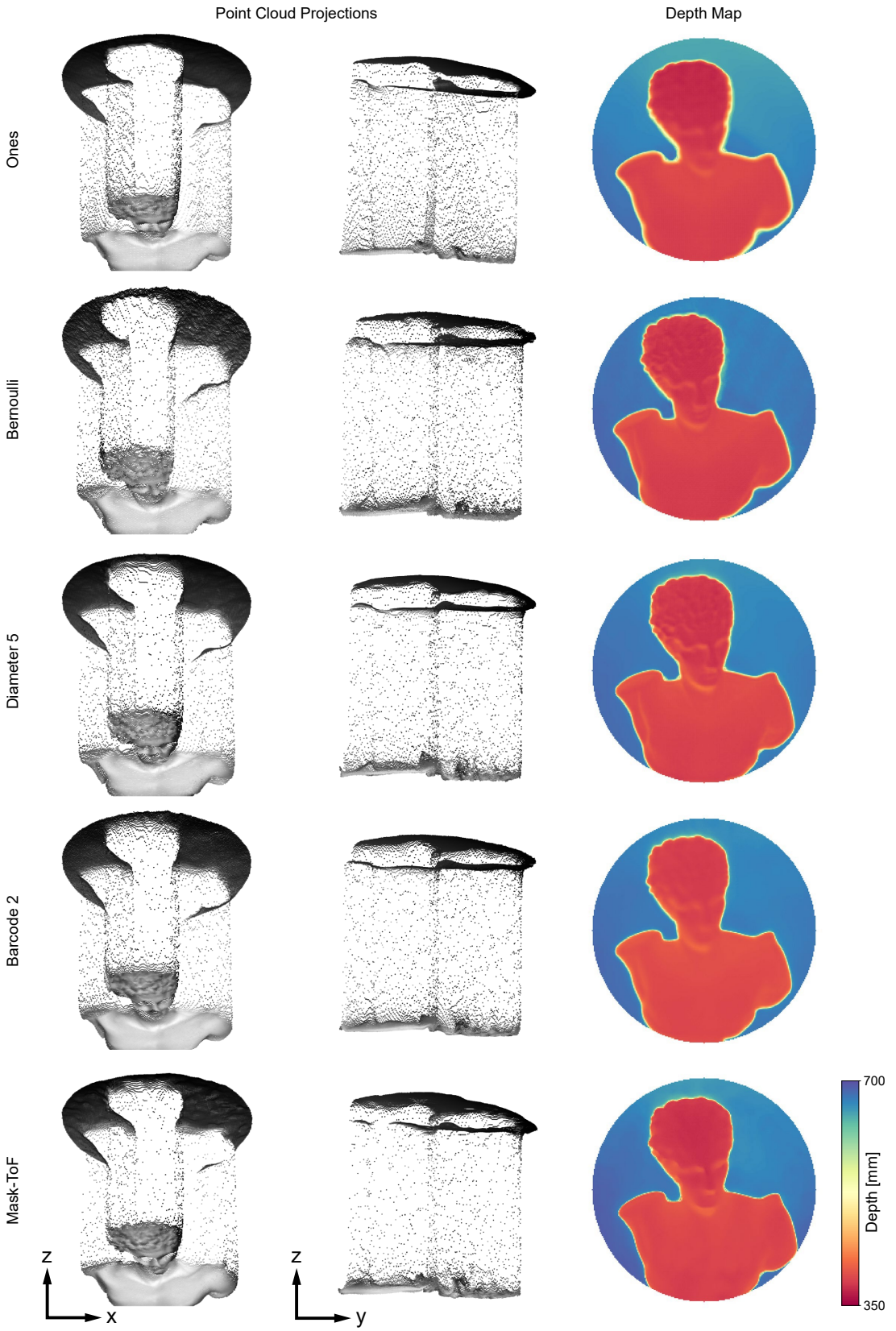


Figure 18: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Hermes*, a replica plaster bust.

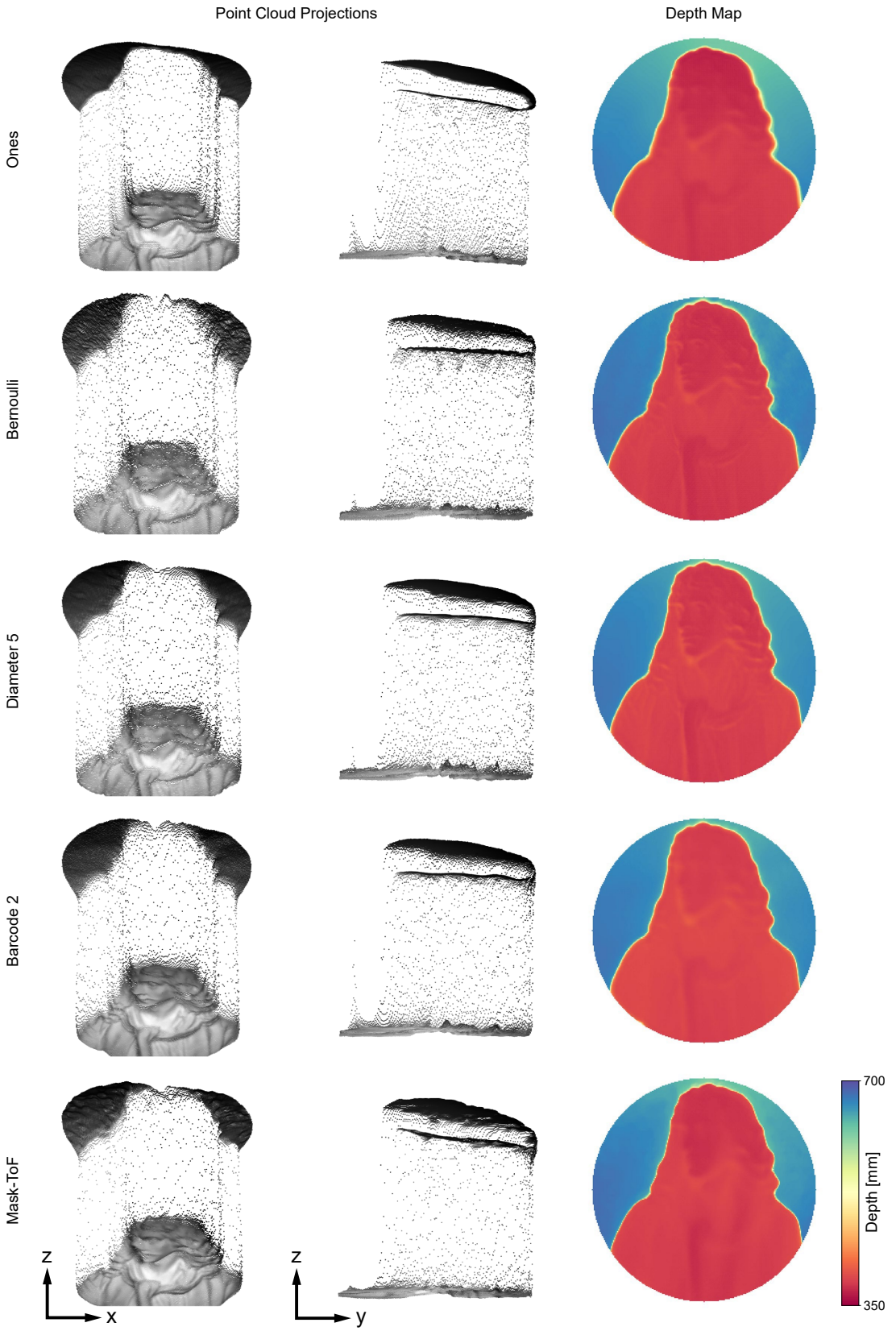


Figure 19: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Moliere*, a replica plaster bust.

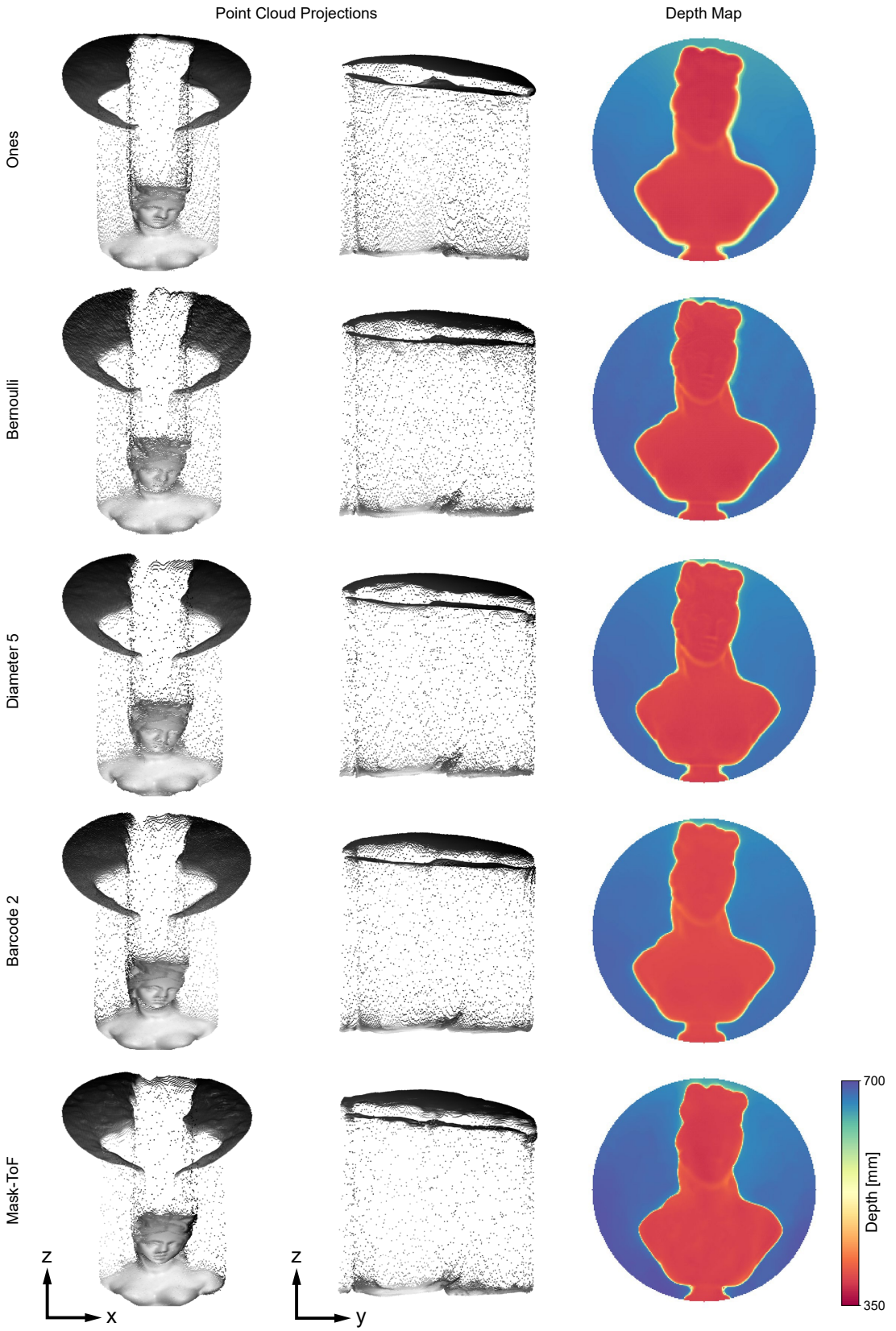


Figure 20: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Aphrodite*, a replica plaster bust.

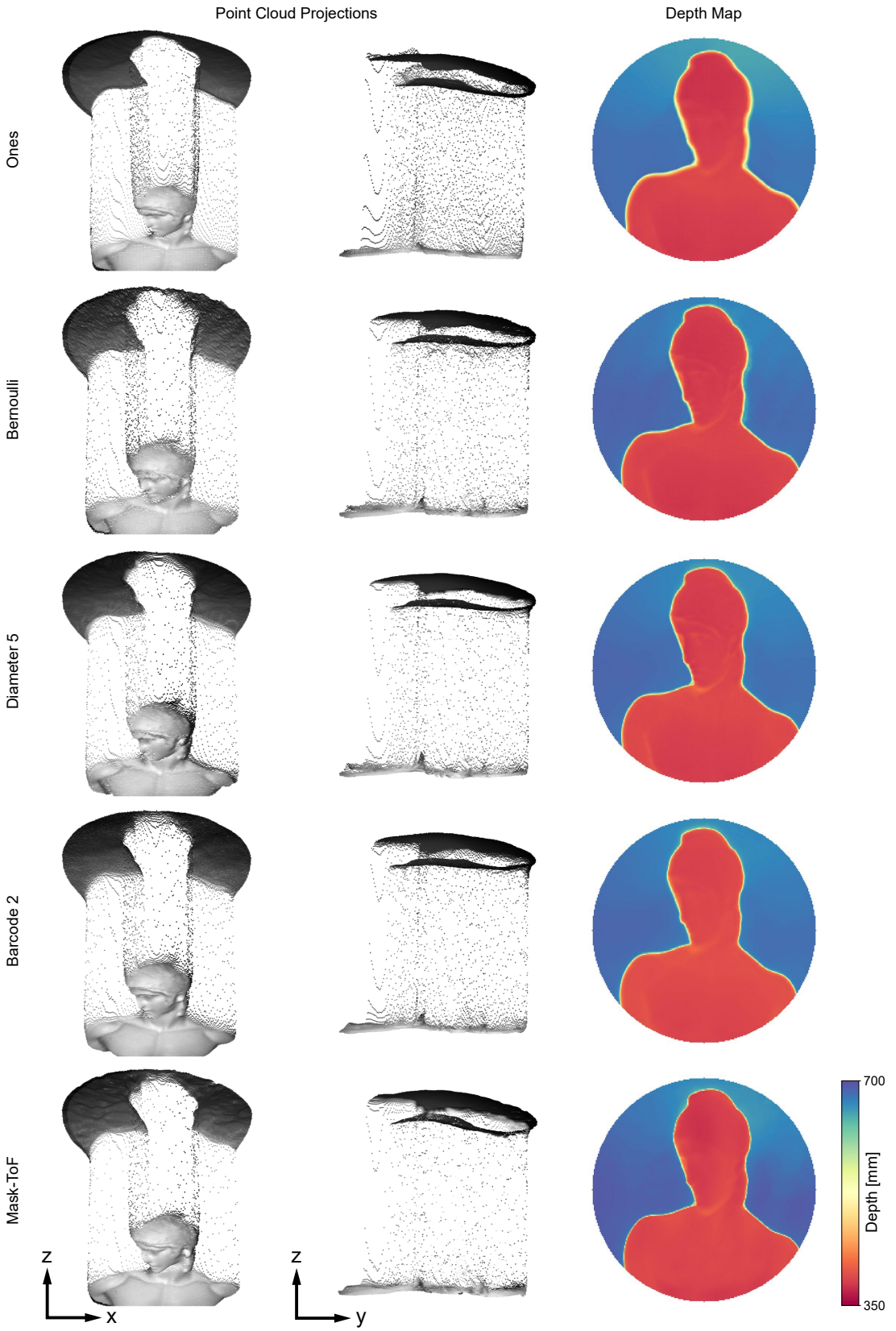


Figure 21: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Ares*, a replica plaster bust.

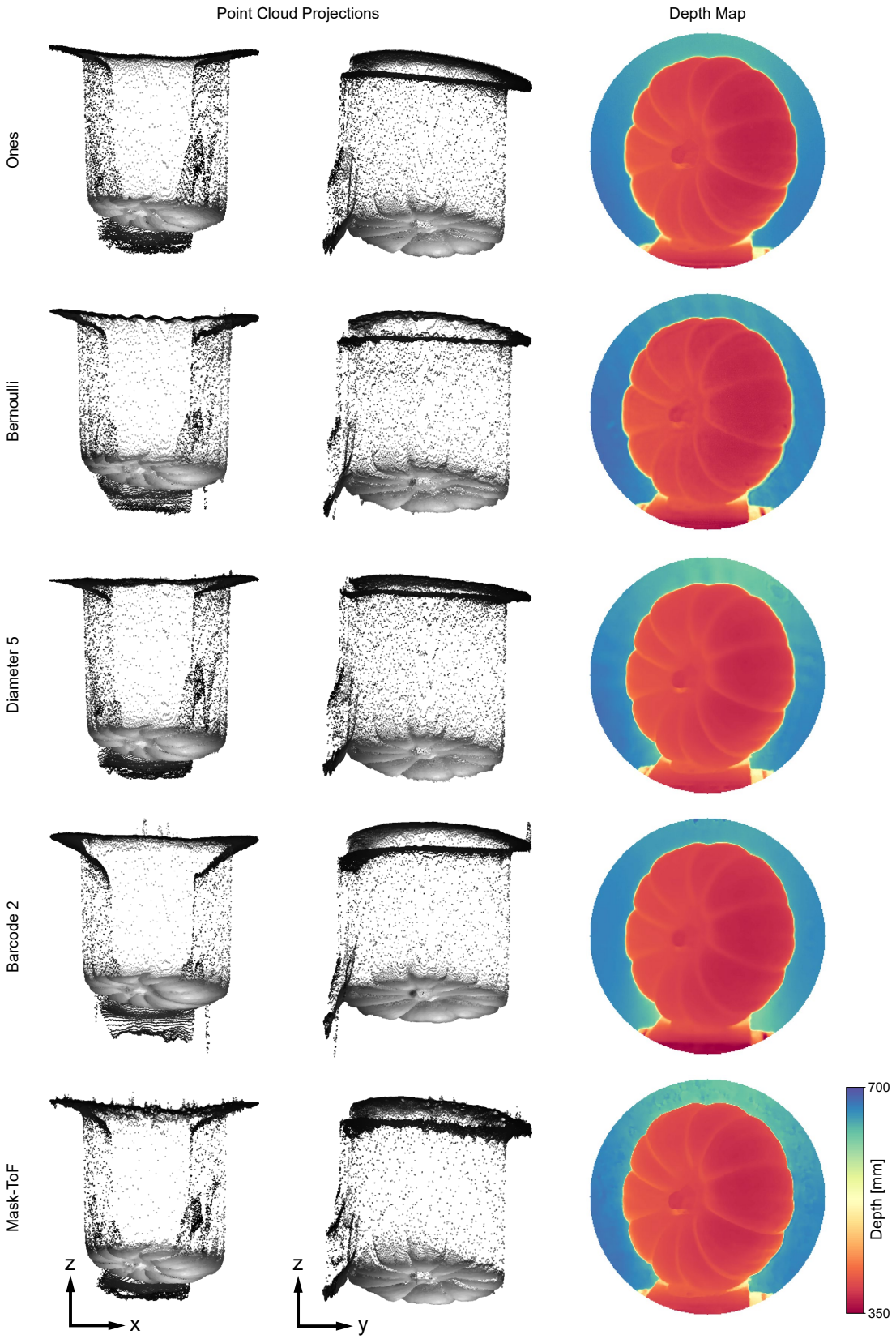


Figure 22: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Pumpkin*.

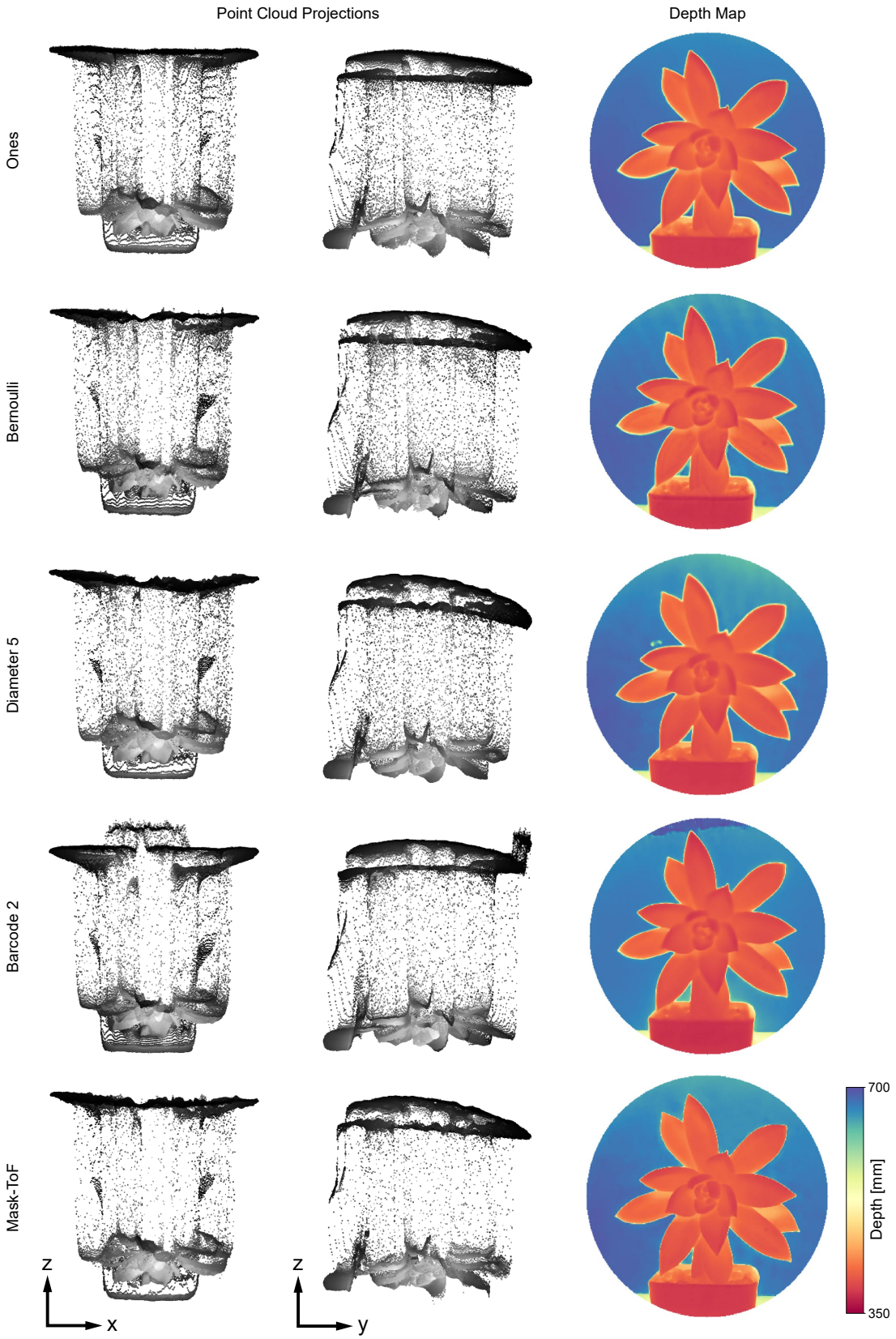


Figure 23: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Plant*.

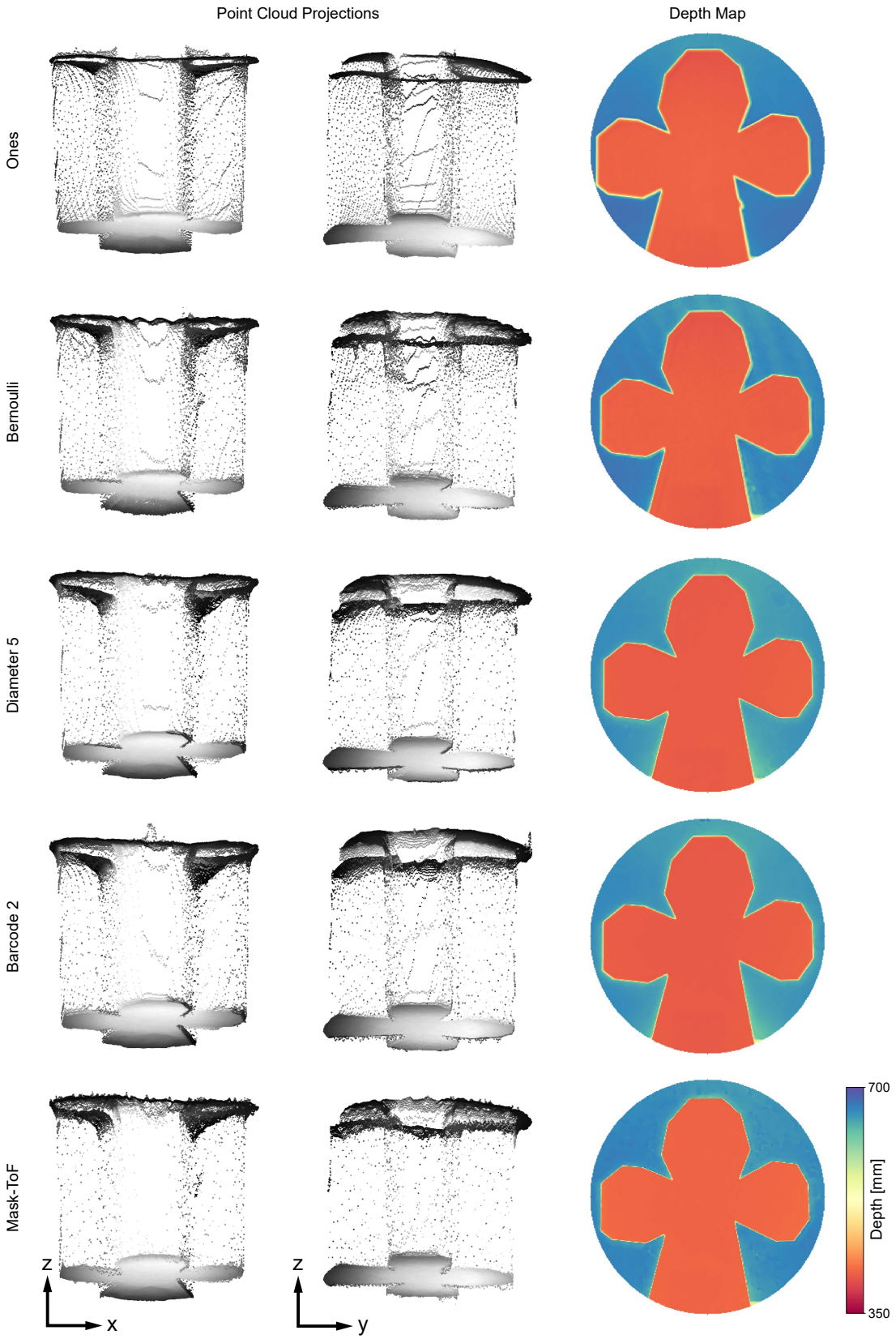


Figure 24: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Club*, a flat cardstock target.

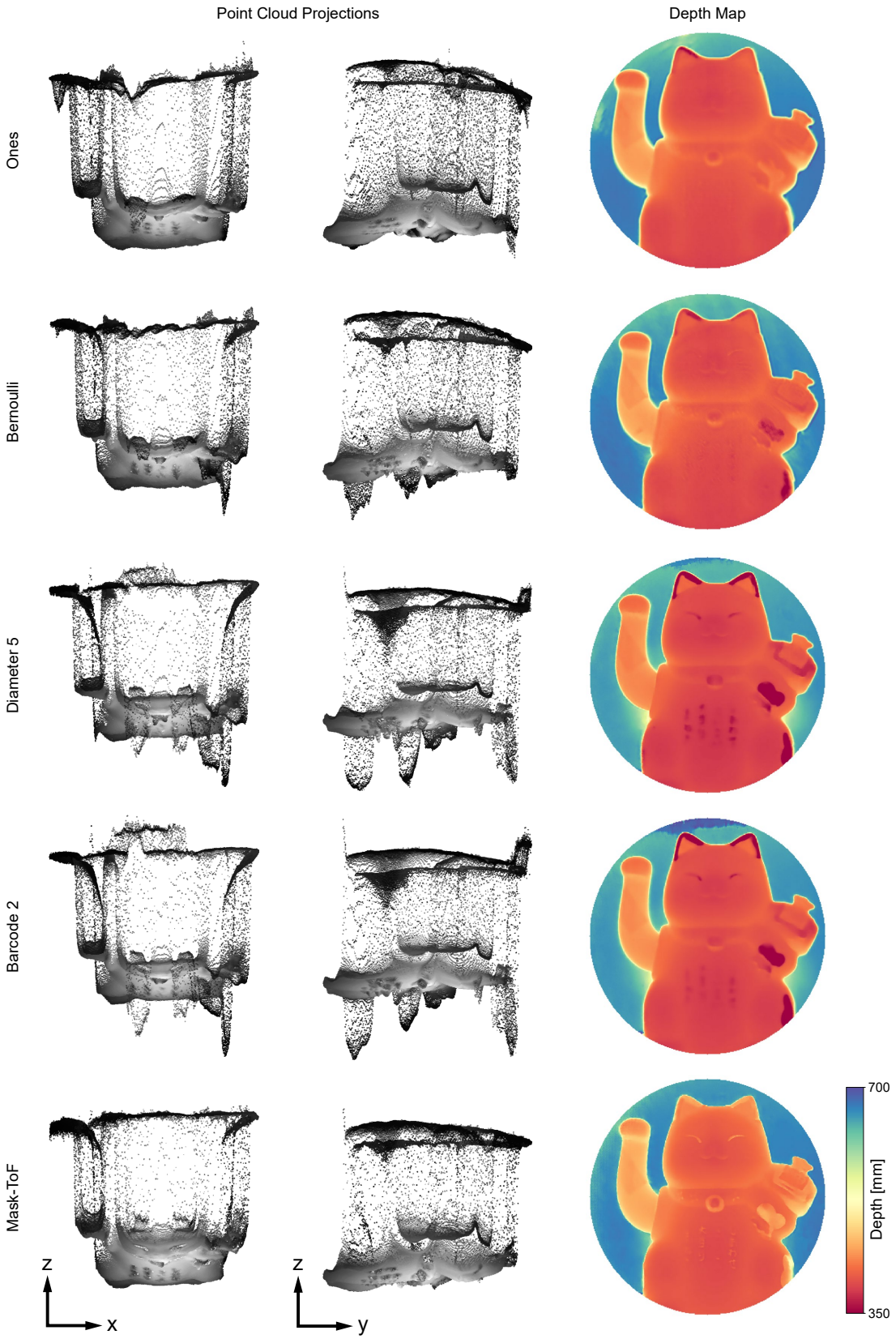


Figure 25: Reconstructed depth maps and point cloud projections for optimized and naïve mask designs from real collected data; test sample *Cat*. The cat's ears, hammer, right leg, and detailing contain sections covered in reflective paint; elsewhere it is matte beige plastic.