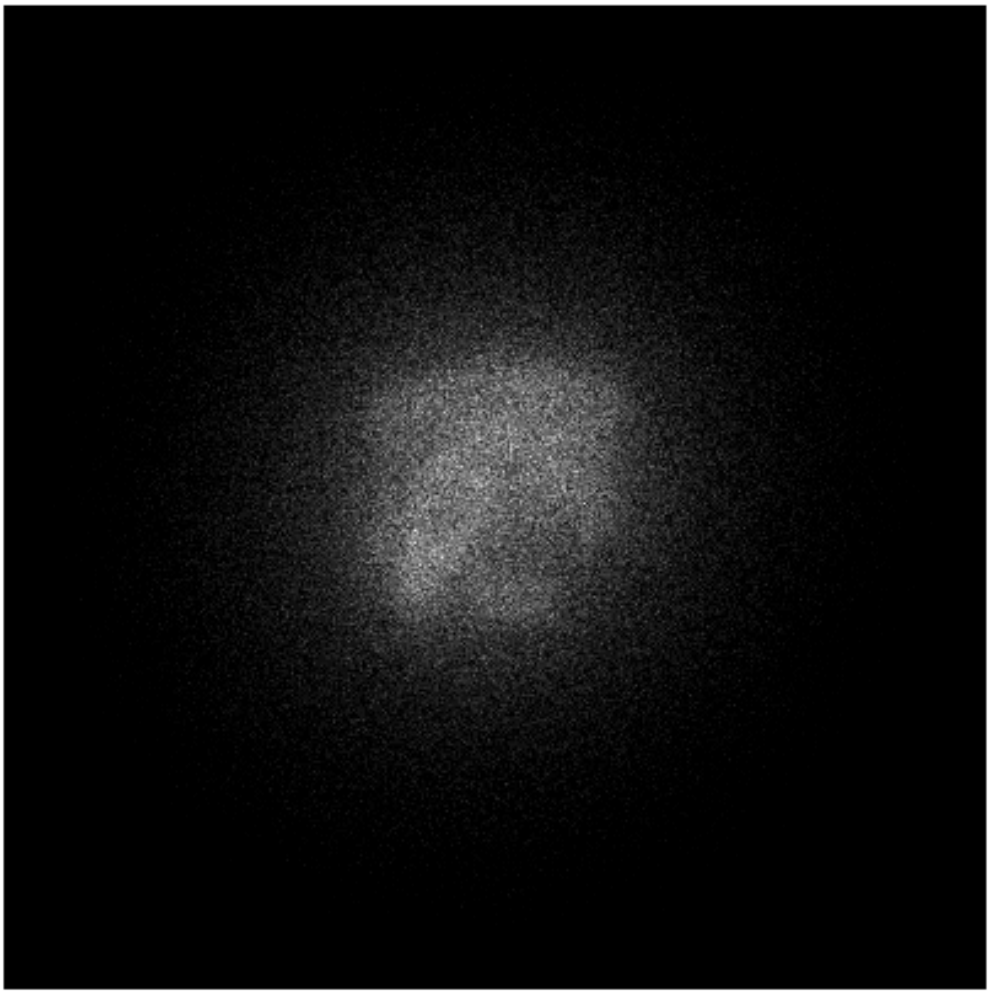
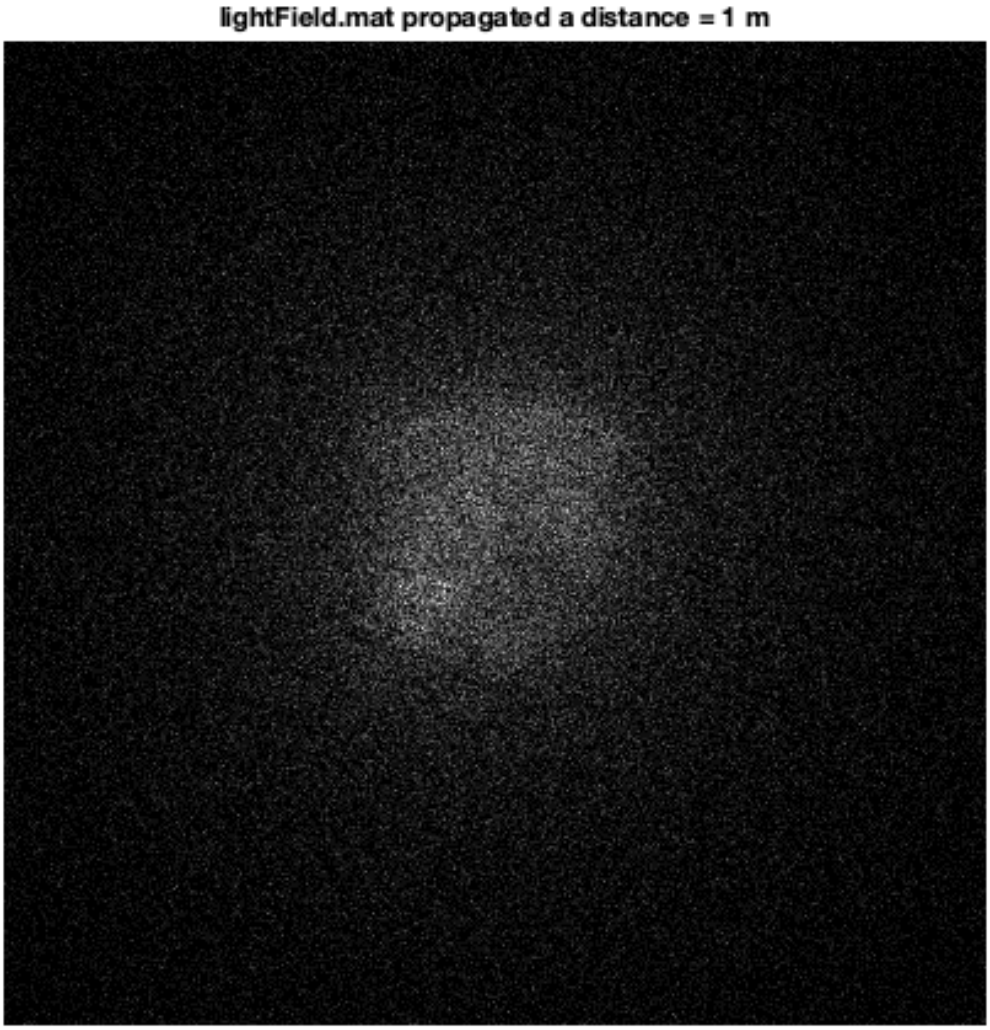
```
% Simulating a Hologram
% Distance = meters; Angle = radians

load('lightField.mat');
width = 0.015; % sensor width parameter for hand-tuning
pixels = 800; % # of pixels parameter for hand-tuning

img = rays2img(rays(1, :), rays(3, :), width, pixels);
figure;
imshow(img);
title("lightField.mat with sensor width = " + width + " m");
exportgraphics(gca, 'light_field.png');

% We are unable to discern the object that generated the rays and cannot
% recover a sharp image by increasing/decreasing neither the width of the
% sensor nor the number of pixels, because the rays are traveling freely in
% space and will only continue to disperse without a lense to focus them
% back together in order to create a sharp image.

d = 1; % d > 0

Md = [1,     d,      0,      0;
      0,     1,      0,      0;
      0,     0,      1,      d;
      0,     0,      0,      1];

rays_out = Md * rays;
img_propagated = rays2img(rays_out(1, :), rays_out(3, :), width, pixels);
figure;
imshow(img_propagated);
title("lightField.mat propagated a distance = " + d + " m");
exportgraphics(gca, 'light_field_propagated.png');

% The rays become even more dispersed and unclear after the propagation,
% with a larger value of d leading to a more blurry image. There is no
% positive value of d that will create a sharp image in the absence of a
% lens, because a simple propagation through space with a positive distance
% will not cause the rays to converge back to a single point to create a
% clear image. In other words, the only ways to make the image clear is to
% propagate them back to the starting point with a negative value of d or
% refocus the rays via lenses.
```

lightField.mat with sensor width = 0.015 m



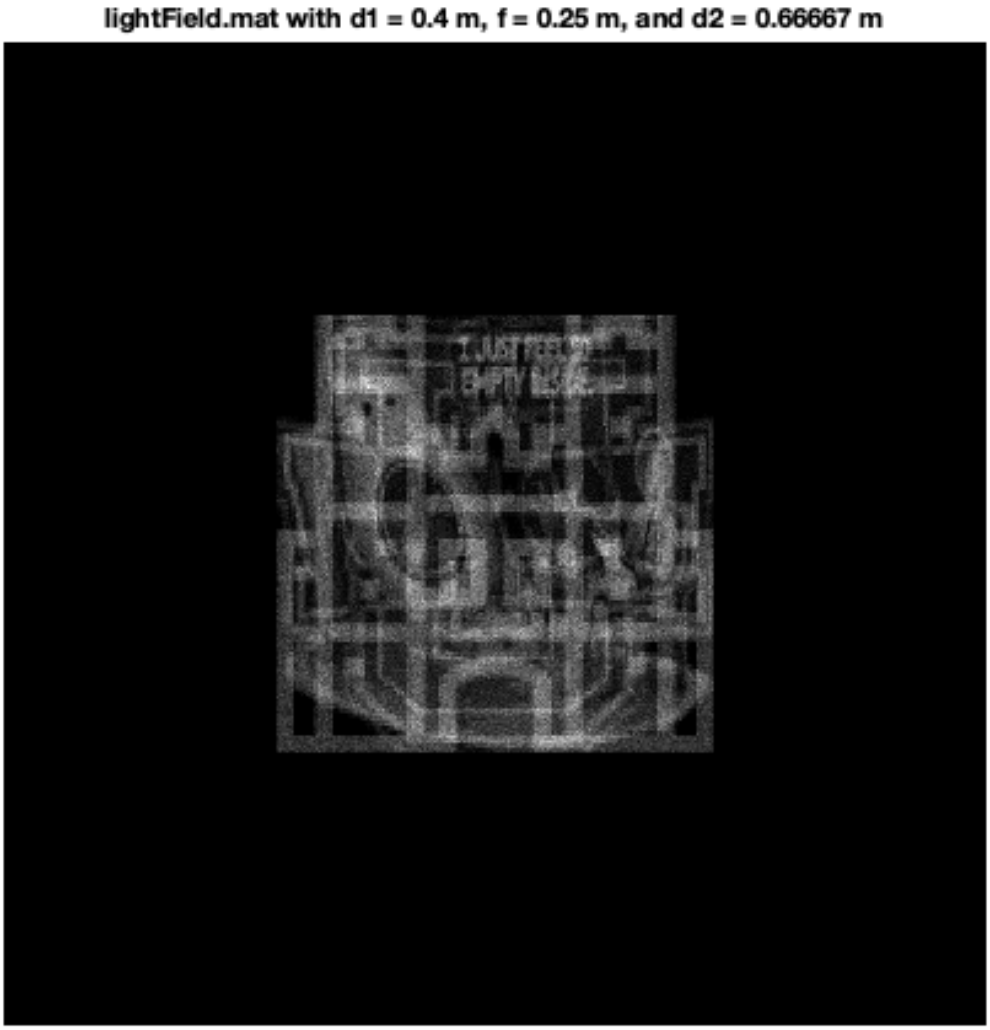


Creating an Image

```
f = 0.25; % constant
d1 = 0.4; % variable

[img_clear, d2] = propagate(rays, width, pixels, f, d1);
figure;
imshow(flip(img_clear, 2));
title("lightField.mat with d1 = " + d1 + " m, f = " + f + ...
    " m, and d2 = " + d2 + " m");
exportgraphics(gca, 'light_field_clear.png');

% Final reported values
disp("d2 = " + d2 + " m");
disp("f = " + f + " m");

% Yes, now we can roughly identify the (although still somewhat blurry)
```

```
% objects that emitted the light rays: an avocado, a person (or a humanoid
% figure), the WashU logo, a building, and the text "I just feel so empty
% inside". This is because the rays are properly propagated with the
% correct value of d1 and a matching combination of d2 and f, allowing the
% 3 million rays to converge to form the visible image.
```

```
d2 = 0.66667 m
f = 0.25 m
```



Published with MATLAB® R2023a