

Overlay a Canadian grid onto images from the Camera Tripod.

```
In[1]:= orig1 = ImageResize[  
    Import["/Volumes/ProjectLibrary/900622.PelagicBenthic/CameraTripod/  
    TankTestSargassoTripod/2013_02_12/Pulse_20130212_00034.JPG"],  
    1000];
```

```
In[2]:= Show[orig1, ImageSize -> Large]
```

Out[2]=



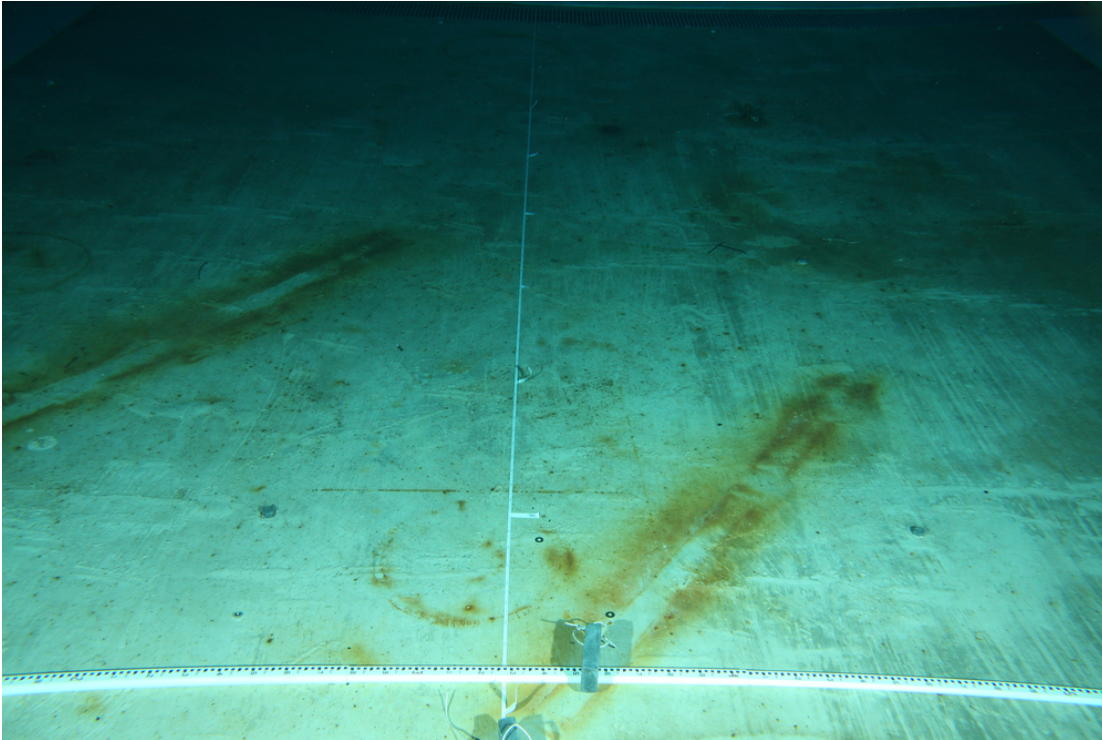
```
In[3]:= ImageDimensions[orig1]
```

Out[3]= { 1000, 667 }

```
In[4]:= orig2 = ImageResize[  
    Import["/Volumes/ProjectLibrary/900622.PelagicBenthic/CameraTripod/  
    TankTestSargassoTripod/2013_02_12/Pulse_20130212_00031.JPG"],  
    1000];
```

In[5]:= Show[orig2, ImageSize → Large]

Out[5]=



Write functions to implement the equations in the Wakefield and Genin paper, “The use of a Canadian (perspective) grid in deep-sea photography.” Note that my functions must have angles input in radians, where the paper uses angles in degrees.

Equation 4.

$$\text{In[6]:= } \text{ba}[\alpha_, \theta_, \text{ON}_-, \text{BA}_-, \text{bp}_-] := \text{bp} \left(1 + \frac{\text{Tan}[\theta - \text{ArcTan}[\text{ON} / (\text{BA} + \text{ON} \text{Cot}[\theta + \alpha / 2])]}{\text{Tan}[\alpha / 2]} \right)$$

Check with example on page 474.

$$\text{In[7]:= } \text{bq} = \text{ba}\left[31 \frac{\text{Pi}}{180}, 30 \frac{\text{Pi}}{180}, 100, 20, 12.5\right]$$

Out[7]= 4.37752

Equation 7.

$$\text{In[8]:= } \text{Shb}[\alpha_, \beta_, \theta_, \text{ON}_-, \text{ef}_-] := \frac{\text{ef} \text{Sin}[\theta + \alpha / 2]}{2 \text{ON} \text{Tan}[\beta / 2]}$$

Check with example on page 474.

$$\text{In[9]:= } \text{shb} = \text{N@Shb}[31 \text{ Degree}, 42 \text{ Degree}, 30 \text{ Degree}, 100, 36]$$

Out[9]= 0.334455

$$\text{In[10]:= } \text{bk} = \text{shb} \cdot 20$$

Out[10]= 6.68909

Equation 10.

$$\text{In}[11]:= \text{Shp}[\beta_, \theta_, \text{ON}_, \text{ef}_] := \frac{\text{ef Sin}[\theta]}{2 \text{ON Tan}[\beta / 2]}$$

Check with example on page 474.

```
In[12]:= shp = N@Shp[42 Degree, 30 Degree, 100, 36]
```

```
Out[12]= 0.234458
```

```
In[13]:= jp = shp 20
```

```
Out[13]= 4.68916
```

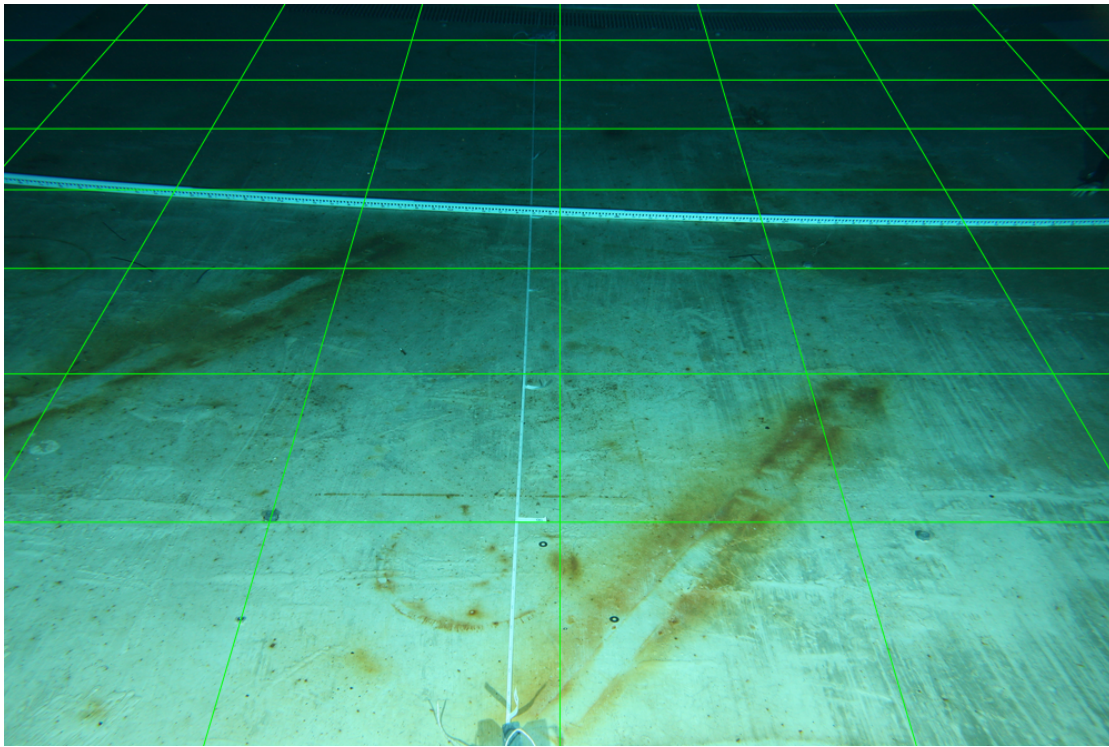
Create a function to overlay a Canadian grid on a given image.

```
In[14]:= canGrid[image_, gsize_, numh_, numv_, α_, β_, θ_, ON_, color_: Green] :=
Module[{line, lines = {}, imageh, i, liney, shb, shp},
  imageh = ImageAspectRatio[image];
  (* calculate horizontal lines *)
  For[i = 1, i <= numh, i++,
    liney = ba[α, θ, ON, i gsize, imageh / 2];
    line = {{0, liney}, {1, liney}};
    AppendTo[lines, line];
  ]; (* end For *)
  (* calculate meridian lines *)
  shb = Shb[α, β, θ, ON, 1];
  shp = Shp[β, θ, ON, 1];
  liney =  $\frac{\text{imageh}}{2} \frac{\text{shb}}{\text{shb} - \text{shp}}$ ; (* find y of vanishing point *)
  For[i = -numv, i <= numv, i++,
    line = {{0.5 - i gsize shb, 0}, {0.5, liney}};
    AppendTo[lines, line];
  ]; (* end For *)
  ImageCompose[image, Graphics[{color, Line[lines]}], PlotRangePadding -> None,
    PlotRange -> {{0, 1}, {0, imageh}}, PlotRangeClipping -> True]
] (* end Module *)
```



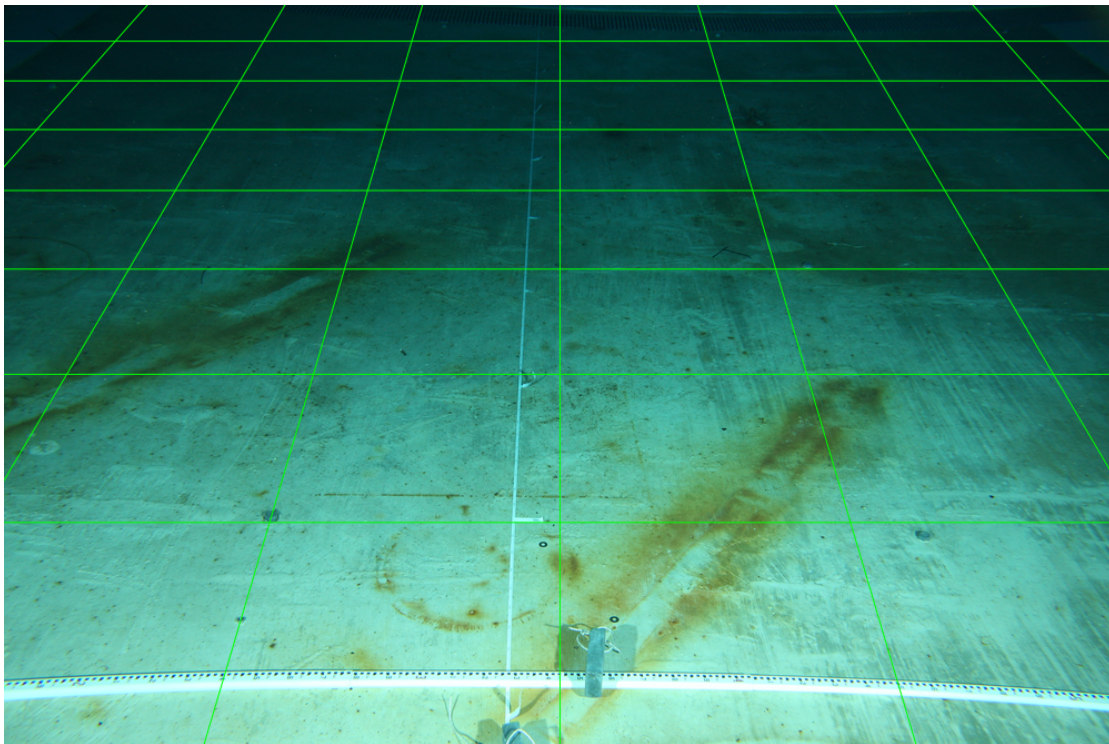
```
In[15]:= canGrid[orig1, 1, 7, 3, 34 Degree, 48 Degree, 31 Degree, 2.6]
```

```
Out[15]=
```



```
In[16]:= canGrid[orig2, 1, 7, 3, 34 Degree, 48 Degree, 31 Degree, 2.6]
```

```
Out[16]=
```



The grids look great but their lines are straight where the scale stick, which is actually straight, appears curved. There is lens distortion caused by the flat viewport, but we can fix that by writing a function to

correct for the pincushion distortion.

```
In[17]:= pinCor[image_] :=  
  ImageForwardTransformation[image, {pt = # - { $\frac{1}{2}$ ,  $\frac{\text{ImageAspectRatio[image]}}{2}$ };  
    pt (1 + -0.23 Norm@pt2) + { $\frac{1}{2}$ ,  $\frac{\text{ImageAspectRatio[image]}}{2}$ }} &]  
  
In[18]:= cor1 = pinCor[orig1]
```

Out[18]=



Now straight objects in the image appear straight, but we should also get rid of as much of the black border as possible.

```
In[19]:= ImageDimensions[cor1]  
  
Out[19]= {1000, 667}
```

```
In[20]:= cor1crop = ImageCrop[cor1]
```

```
Out[20]=
```

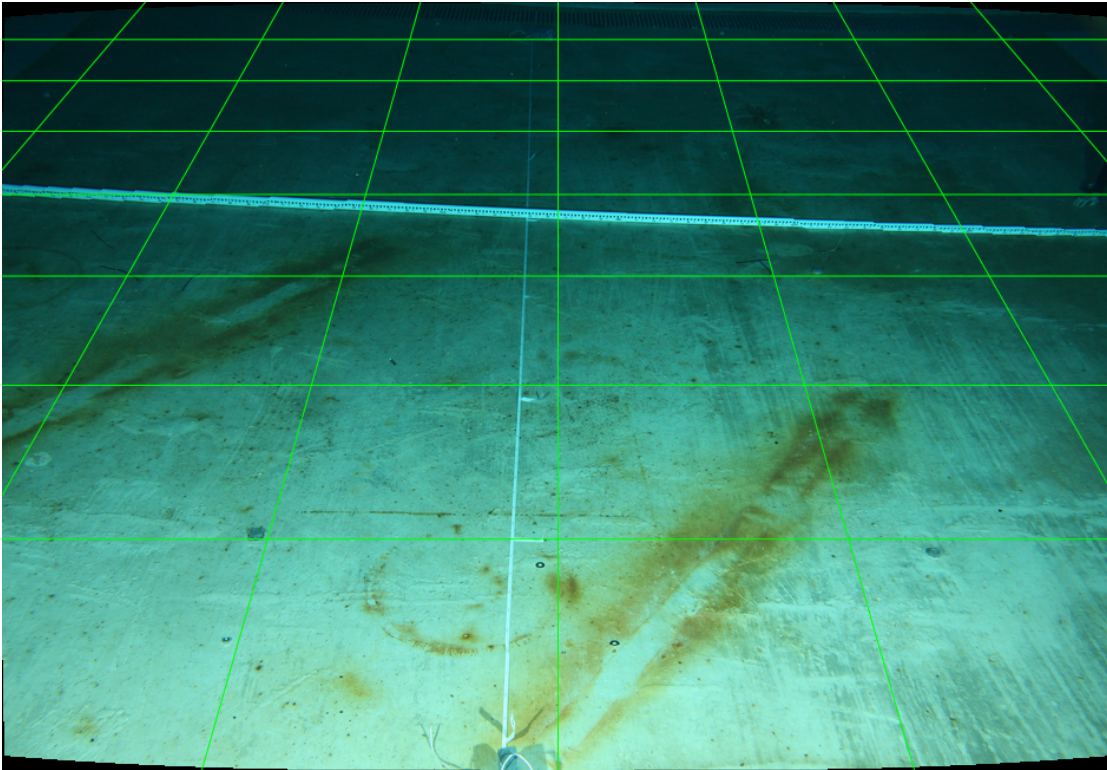


```
In[21]:= ImageDimensions[cor1crop]
```

```
Out[21]= { 942, 651 }
```

Apply the grid lines to the corrected image.

In[22]:= `canGrid[cor1crop, 1, 7, 3, 34 Degree, 48 Degree, 31 Degree, 2.6]`

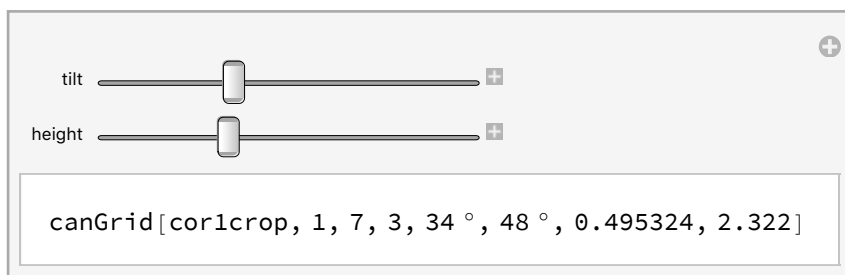


Out[22]=

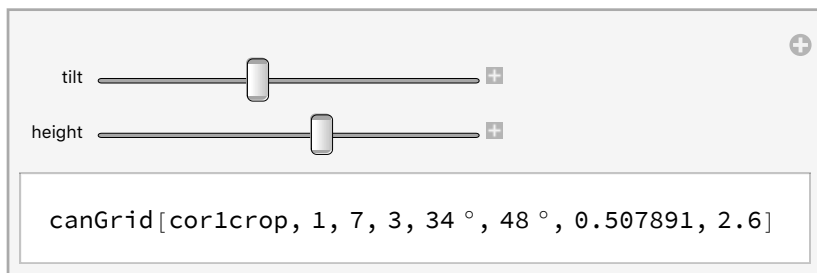
Now both the scale stick and the Canadian grid lines are straight, but the distance between grid lines, set at 1 meter, doesn't match the markings on the ruler stick or tape. We can interactively adjust the height and tilt of the camera to find values that correct the error.

In[43]:= `Manipulate[canGrid[cor1crop, 1, 7, 3, 34 Degree, 48 Degree, tilt Degree, height],
{tilt, 31}, 25, 35], {{height, 2.6}, 2, 3}]`

Out[43]=



In[24]:=



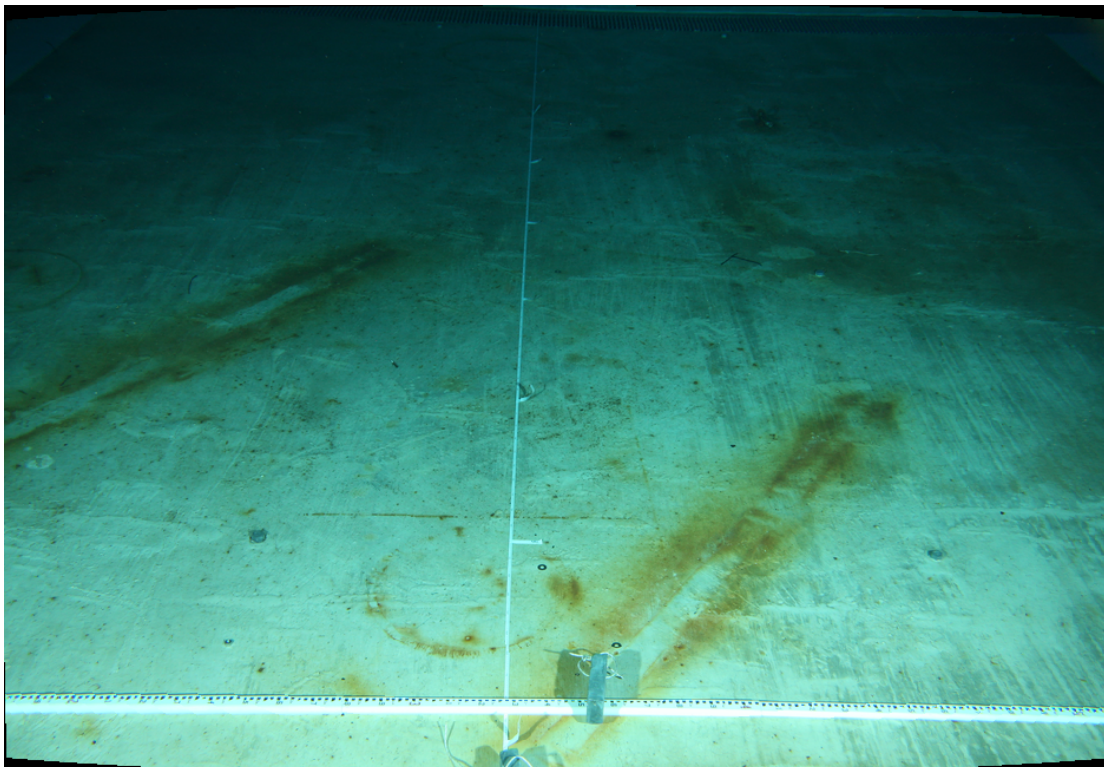

```
In[25]:= pinCor[orig2]
```

Out[25]=



```
In[26]:= cor2crop = ImageCrop[pinCor[orig2]]
```

Out[26]=



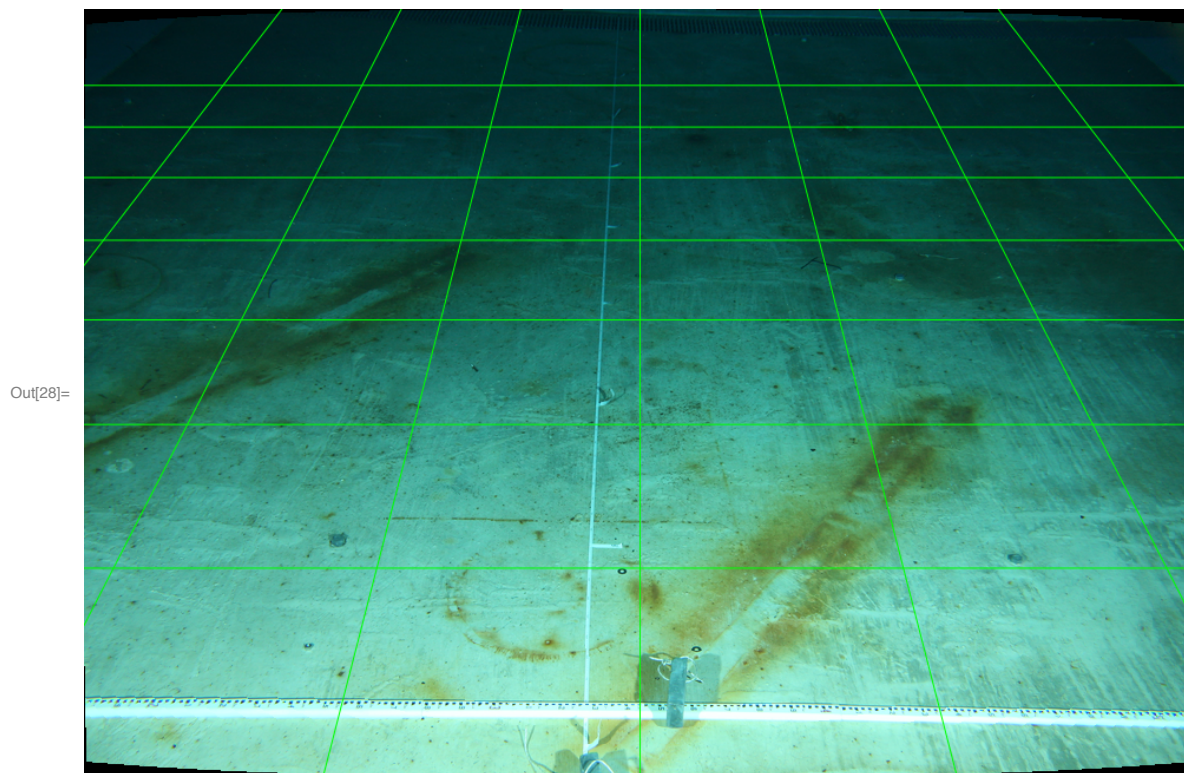
```
In[27]:= Manipulate[canGrid[ImageAdjust[cor2crop], 1, 7, 3, 34 Degree, 48 Degree,
    tilt Degree, height], {{tilt, 31}, 25, 35}, {{height, 2.6}, 2, 3}]
```

Out[27]=

canGrid[ImageAdjust[cor2crop], 1, 7, 3, 34 °, 48 °, 30 °, 2.6]

ImageAdjust: Expecting an image or graphics instead of cor2crop.

```
In[28]:= canGrid[cor2crop, 1, 7, 3, 34 Degree, 48 Degree, 30.5 Degree, 2.9]
```



```
In[29]:= DynamicImage[canGrid[cor2crop, 1, 7, 3, 34 Degree, 48 Degree, 30.5 Degree, 2.9]]
```

Out[29]= **\$Failed**



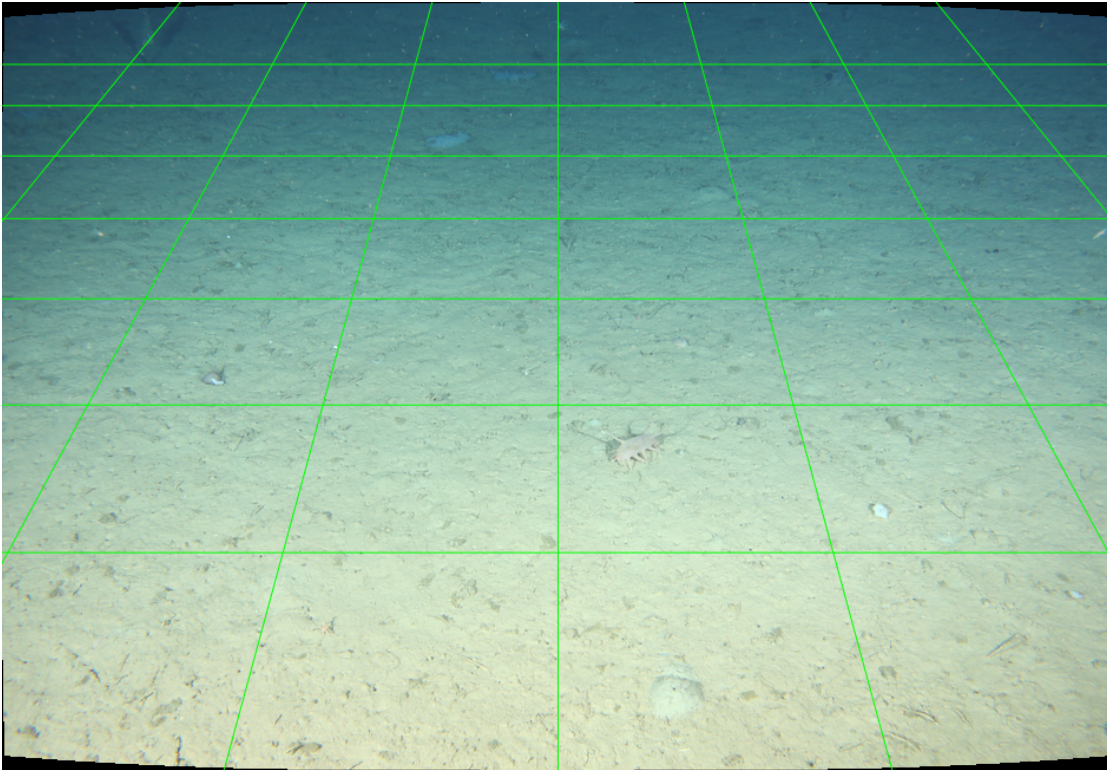
```
In[30]:= staM1 = ImageResize[
    Import["/Volumes/pulse/STORE/Files/PULSE/Pulse\ 67/Pulse\ 67\ Camera\
    Tripod/Far-Field/images/StaM_6701_FF_tripod_20161121_07_52_25.JPG"],
    1000];
```

```
In[31]:= staM1c = ImageCrop[pinCor[staM1]];
```



```
In[32]:= canGrid[ImageAdjust@staM1c, 1, 7, 3, 34 Degree, 48 Degree, 30.5 Degree, 2.75]
```

```
Out[32]=
```

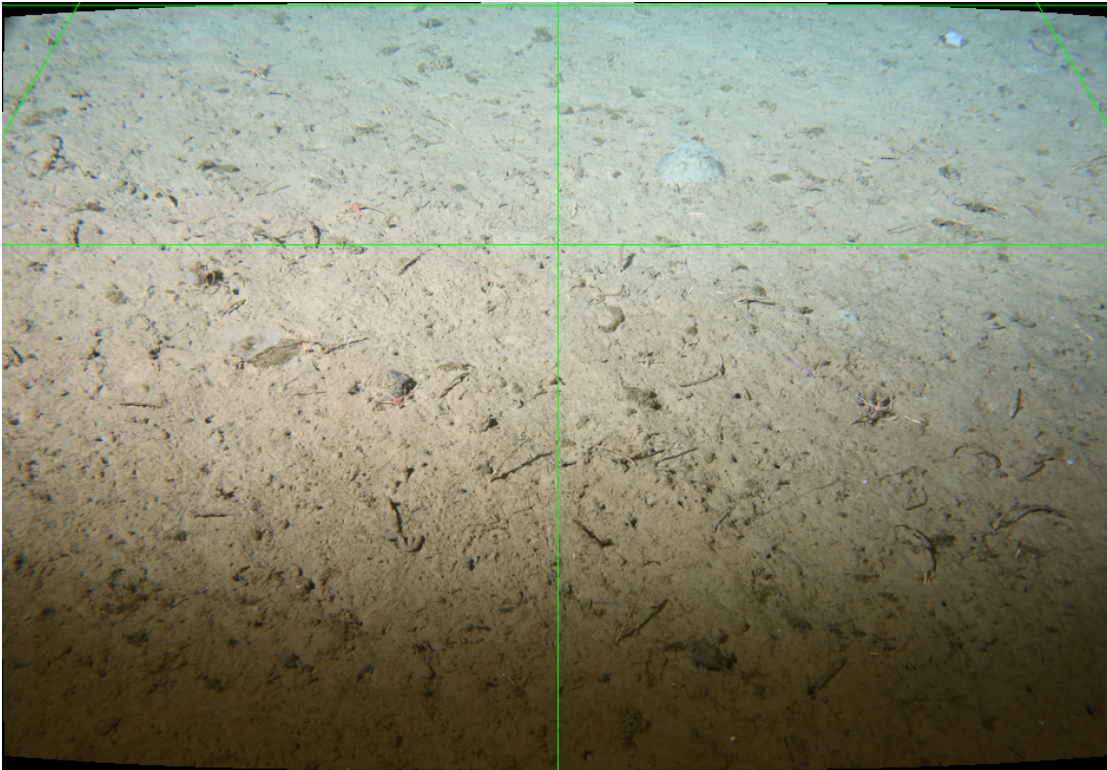


```
In[33]:= staM2 = ImageResize[
  Import["/Volumes/pulse/STORE/Files/PULSE/Pulse\ 67/Pulse\ 67\ Camera\
    Tripod/Near-Field/images/StaM_6701_NF_tripod_20161121_07_52_20.JPG"],
  1000];
```

```
In[34]:= staM2c = ImageCrop[pinCor[staM2]];
```


In[35]:= **canGrid[ImageAdjust@staM2c, 1, 7, 3, 34 Degree, 48 Degree, 39 Degree, 1.12]**

Out[35]=



In[36]:= /

