bias_and_dark

May 7, 2020

1 Processing Science Images

1.1 Bias and Dark: Written by Maria Funcich

[1]:
```
import numpy as np
import astropy
import ccdproc
from ccdproc import CCDData, combiner
from astropy import units as u
import matplotlib.pyplot as plt
from matplotlib.colors import LogNorm
```

2 Bias

2.0.1 ⋆ Collecting and Loading the Bias frames

[2]:
```
# creating image list
images = ccdproc.ImageFileCollection(".")

# filtering the Bias images and adding them to a list 'bias_img'
##print(images.files_filtered(PICTTYPE = 2))
bias_img = (images.files_filtered(PICTTYPE = 2))

# printing list of the bias images
print("List of Bias Images")
for fn in bias_img:
    print(fn)

# reading the bias frames in ADU and defining them into list 'biases'
biases = [CCDData.read(fn, unit = "adu") for fn in bias_img]
```

```
List of Bias Images
Bias_-4.82C_00001658.fit
Bias_-4.82C_00001663.fit
Bias_-4.82C_00001667.fit
```

```
Bias_-4.82C_00001670.fit
Bias_-4.82C_00001671.fit
Bias_-4.82C_00001672.fit
Bias_-4.82C_00001675.fit
Bias_-4.82C_00001676.fit
Bias_-4.82C_00001677.fit
Bias_-4.82C_00001678.fit
Bias_-4.82C_00001679.fit
Bias_-4.82C_00001681.fit
Bias_-4.82C_00001682.fit
Bias_-4.82C_00001683.fit
Bias_-4.82C_00001684.fit
Bias_-4.82C_00001688.fit
Bias_-4.82C_00001691.fit
Bias_-4.82C_00001692.fit
Bias_-4.82C_00001722.fit
Bias_-4.82C_00001723.fit
Bias_-4.82C_00001724.fit
Bias_-4.82C_00001725.fit
Bias_-4.82C_00001726.fit
Bias_-4.82C_00001727.fit
Bias_-4.82C_00001728.fit
Bias_-4.82C_00001729.fit
Bias_-4.82C_00001730.fit
Bias_-4.82C_00001731.fit
```

2.0.2 ⋆ Checking individual Bias Frames

Checking each bias frame in DS9: In terminal: "ds9 -zscale Bias_* &"

- First check is to identify if they are all bias frames. This is true
- All frames appear to have more light coming in from the left side of the image
- Checking through our science images, I can see the same trend occuring therefore the bias frames are useable

Checking the statisitics for the first frame

[3]:
```
print("Pixel Value (ADU): All Bias Images", "\n", biases, "\n") # prints list␣
↪of pixel values for all the bias images
print("Pixel Value (ADU): First Bias Image", "\n", biases[0], "\n") # prints␣
↪the pixel values of the first bias image
print("Pixel Value (ADU): First Bias Image, First Column:", "\n", biases[0][:
↪,0], "\n") # print the counts of the first column of the first bias image
```

```
Pixel Value (ADU): All Bias Images
 [CCDData([[141, 147, 134, …, 111, 110, 120],
           [128, 135, 135, …, 115, 107, 104],
```

```
        [135, 122, 122, …, 105, 110, 114],
        …,
        [142, 139, 135, …, 114, 111, 110],
        [133, 144, 134, …, 105, 101, 105],
        [135, 124, 138, …, 105, 106, 105]]), CCDData([[144, 134, 143, …,
107, 100, 108],
        [137, 145, 143, …, 105, 110, 109],
        [134, 134, 139, …, 112, 109, 101],
        …,
        [137, 127, 134, …, 105, 110, 103],
        [135, 135, 135, …, 115,  99, 113],
        [132, 136, 136, …, 116, 116, 107]]), CCDData([[121, 133, 134, …,
104, 102, 115],
        [130, 136, 120, …, 116,  99,  97],
        [132, 132, 118, …, 104, 102, 100],
        …,
        [126, 132, 131, …, 105, 101, 110],
        [137, 123, 122, …, 107,  99, 101],
        [119, 126, 130, …, 102,  97,  98]]), CCDData([[130, 134, 139, …,
103,  94,  99],
        [127, 128, 134, …,  97, 106, 101],
        [136, 135, 130, …, 101,  95,  93],
        …,
        [122, 133, 124, …, 102, 105,  97],
        [134, 123, 129, …,  92,  93, 107],
        [128, 127, 134, …, 113,  97,  98]]), CCDData([[136, 134, 131, …,
107, 105, 100],
        [140, 135, 133, …, 105, 100, 103],
        [131, 129, 137, …, 113,  97, 100],
        …,
        [138, 124, 122, …,  97, 106, 104],
        [124, 129, 131, …,  99, 101, 104],
        [144, 130, 122, …, 104, 107,  97]]), CCDData([[137, 126, 138, …,
110,  99, 107],
        [135, 131, 124, …, 101, 100,  99],
        [132, 117, 123, …, 101,  99, 115],
        …,
        [118, 133, 133, …, 107,  99,  82],
        [128, 135, 127, …, 105, 106, 107],
        [123, 137, 124, …,  97, 111, 109]]), CCDData([[137, 135, 136, …,
102, 107, 113],
        [145, 132, 131, …, 105, 115, 122],
        [143, 136, 141, …, 111, 103, 109],
        …,
        [122, 125, 136, …, 111, 110, 117],
        [130, 124, 135, …, 100, 116, 105],
        [138, 139, 134, …, 105, 103,  99]]), CCDData([[144, 134, 145, …,
101, 103, 107],
```

```
         [135, 129, 130, …, 105, 104, 100],
         [131, 136, 139, …, 104, 105, 108],
         …,
         [121, 130, 127, …, 104, 103, 101],
         [133, 132, 133, …, 112,  98, 116],
         [135, 147, 132, …, 110, 119, 102]]), CCDData([[129, 134, 118, …,
99,  99,  91],
         [126, 136, 137, …,  95, 101,  92],
         [127, 123, 134, …, 100,  95, 100],
         …,
         [127, 125, 118, …, 100,  97, 102],
         [124, 137, 116, …, 108,  89,  92],
         [124, 112, 125, …,  94, 107, 104]]), CCDData([[138, 117, 142, …,
111, 119, 105],
         [136, 150, 140, …, 109, 121, 106],
         [139, 142, 135, …, 109, 105, 100],
         …,
         [152, 133, 144, …, 100, 107, 106],
         [141, 124, 138, …, 108, 121, 120],
         [137, 146, 128, …, 105, 109, 102]]), CCDData([[135, 144, 133, …,
103, 102, 111],
         [125, 132, 123, …, 109,  97, 108],
         [122, 124, 144, …, 101, 114,  98],
         …,
         [129, 133, 138, …, 111, 108, 114],
         [128, 133, 125, …, 101,  93, 104],
         [121, 125, 144, …, 113,  90, 106]]), CCDData([[135, 124, 115, …,
98,  92,  97],
         [134, 130, 138, …, 107,  95,  86],
         [131, 134, 116, …,  99,  93,  96],
         …,
         [125, 122, 118, …,  92,  88, 100],
         [128, 121, 129, …, 103,  97,  97],
         [125, 117, 135, …,  91,  89,  97]]), CCDData([[142, 133, 125, …,
115, 117,  97],
         [125, 133, 141, …, 108, 102,  86],
         [133, 134, 133, …, 101, 100, 103],
         …,
         [127, 134, 111, …, 100, 112,  96],
         [132, 129, 126, …,  97, 103, 107],
         [135, 143, 124, …,  97,  95, 108]]), CCDData([[106, 116, 127, …,
113,  95,  94],
         [130, 123, 127, …, 105, 104, 109],
         [127, 120, 124, …, 112,  86,  92],
         …,
         [119, 122, 124, …,  90,  91,  94],
         [129, 125, 128, …, 111,  97,  94],
         [123, 126, 124, …, 103,  97,  97]]), CCDData([[135, 119, 125, …,
```

```
97,  97, 104],
         [128, 124, 132, …,  96, 104, 105],
         [128, 140, 138, …, 109, 110,  95],
         …,
         [128, 126, 130, …, 109, 100, 107],
         [132, 132, 127, …, 116, 109,  97],
         [132, 122, 123, …, 100, 102, 108]]), CCDData([[130, 119, 110, …,
98, 105,  93],
         [137, 123, 133, …, 108,  96,  97],
         [135, 138, 127, …, 100,  96,  96],
         …,
         [132, 130, 130, …, 105,  93,  94],
         [125, 127, 136, …,  99, 101, 107],
         [129, 135, 129, …,  99, 102, 103]]), CCDData([[132, 120, 134, …,
102, 104,  98],
         [117, 137, 135, …,  97, 102, 108],
         [132, 117, 135, …, 107, 100, 104],
         …,
         [126, 128, 121, …,  98,  93, 101],
         [120, 131, 130, …,  95, 102,  96],
         [138, 116, 118, …, 104,  94,  92]]), CCDData([[123, 125, 136, …,
98, 104, 109],
         [128, 120, 129, …, 108, 102,  99],
         [133, 142, 123, …, 103, 104, 103],
         …,
         [114, 130, 114, …, 101, 111,  90],
         [121, 141, 134, …, 106, 103, 107],
         [133, 125, 127, …,  92, 105,  99]]), CCDData([[126, 129, 131, …,
100, 106, 104],
         [124, 134, 124, …,  94, 108, 101],
         [130, 136, 130, …, 108,  88,  94],
         …,
         [125, 135, 135, …, 109, 104, 105],
         [131, 132, 123, …,  99,  99,  97],
         [121, 121, 127, …, 105, 105, 102]]), CCDData([[137, 129, 128, …,
106, 106, 103],
         [141, 141, 131, …, 103, 109, 102],
         [127, 144, 136, …, 121, 113, 106],
         …,
         [126, 131, 130, …, 115, 104, 115],
         [135, 138, 117, …, 111, 105, 112],
         [125, 140, 137, …, 108, 105, 121]]), CCDData([[139, 129, 136, …,
107,  99, 102],
         [131, 131, 133, …,  98, 102,  99],
         [125, 120, 143, …, 108, 105,  98],
         …,
         [135, 130, 125, …, 101, 103, 103],
         [137, 126, 140, …, 101, 112, 107],
```

```
        [129, 129, 125, …, 107, 109, 108]]), CCDData([[138, 131, 138, …,
107, 112, 103],
        [127, 132, 132, …, 104,  96, 103],
        [134, 137, 124, …,  91, 105, 105],
        …,
        [123, 130, 131, …, 117, 103, 104],
        [134, 123, 134, …, 101,  96, 116],
        [125, 131, 129, …, 106, 103, 108]]), CCDData([[143, 132, 130, …,
98, 104, 118],
        [139, 134, 129, …, 107, 111, 113],
        [142, 135, 141, …, 103, 100, 105],
        …,
        [133, 137, 142, …, 108, 109,  99],
        [134, 118, 133, …, 103, 111, 117],
        [137, 130, 123, …, 110, 114, 109]]), CCDData([[130, 133, 130, …,
115, 108,  99],
        [126, 132, 121, …, 110, 104, 105],
        [142, 127, 139, …, 100, 115, 104],
        …,
        [131, 122, 123, …, 101, 106, 109],
        [124, 131, 129, …, 110,  96, 111],
        [124, 138, 130, …,  99,  96, 113]]), CCDData([[126, 136, 128, …,
101, 101, 107],
        [131, 133, 133, …, 105,  97, 115],
        [129, 127, 137, …, 107,  98, 104],
        …,
        [125, 119, 137, …, 112,  99, 105],
        [129, 134, 129, …, 100, 113,  91],
        [131, 123, 131, …, 107, 110, 109]]), CCDData([[135, 134, 140, …,
99, 104, 119],
        [141, 136, 137, …,  96, 112, 117],
        [138, 133, 136, …, 111, 105, 100],
        …,
        [124, 134, 136, …, 113, 106, 109],
        [132, 146, 139, …, 106, 100, 109],
        [131, 144, 127, …, 116, 107, 117]]), CCDData([[139, 129, 134, …,
106, 101,  96],
        [132, 141, 130, …, 118, 115, 104],
        [141, 137, 136, …, 107, 107, 105],
        …,
        [142, 107, 134, …, 111, 114, 112],
        [137, 134, 140, …, 106, 105, 103],
        [139, 134, 145, …, 114, 110, 108]]), CCDData([[132, 146, 133, …,
117, 103,  96],
        [129, 150, 135, …, 100, 106, 118],
        [135, 136, 145, …, 105, 107, 126],
        …,
        [136, 127, 121, …, 114, 106, 107],
```

```
          [134, 133, 120, …, 112, 107, 106],
          [137, 137, 129, …, 121, 108, 100]])]

Pixel Value (ADU): First Bias Image
 [[141 147 134 … 111 110 120]
 [128 135 135 … 115 107 104]
 [135 122 122 … 105 110 114]
 …
 [142 139 135 … 114 111 110]
 [133 144 134 … 105 101 105]
 [135 124 138 … 105 106 105]]

Pixel Value (ADU): First Bias Image, First Column:
 [141 128 135 135 134 132 130 133 138 133 125 143 130 137 130 139 147 140
 123 133 139 129 138 141 130 136 133 140 139 135 142 140 117 144 140 139
 139 124 129 124 126 130 137 138 130 144 138 130 137 123 144 136 132 137
 135 149 135 145 134 131 151 130 142 133 141 138 131 141 140 144 140 134
 132 131 132 128 130 131 134 126 130 136 130 125 133 128 130 145 135 133
 134 136 140 135 137 135 133 128 140 119 132 128 146 141 130 132 130 127
 137 134 121 128 140 132 127 133 138 139 141 129 130 136 123 126 134 134
 132 131 140 128 128 125 145 137 138 124 129 136 133 135 130 124 132 135
 133 134 134 130 133 134 153 133 133 135 143 128 144 138 135 138 136 138
 140 134 131 131 140 119 136 132 137 133 144 136 135 134 141 129 140 128
 138 129 131 131 138 127 129 123 136 136 136 139 139 128 142 131 133 131
 124 132 132 136 139 138 146 139 132 141 137 131 133 129 134 136 134 145
 134 133 133 129 140 140 136 143 125 144 138 128 127 125 133 113 137 143
 144 125 135 130 138 136 130 130 134 133 143 122 134 148 133 142 128 145
 136 117 142 139 134 130 127 122 130 127 148 142 143 144 131 136 130 131
 139 128 127 131 127 135 134 141 136 136 149 135 134 136 135 131 133 139
 132 130 131 135 134 137 146 138 136 130 129 145 131 137 135 127 134 123
 130 142 138 133 135 135 136 133 130 127 139 134 133 146 124 133 130 132
 136 126 132 128 132 126 136 139 138 131 140 140 136 132 129 132 141 138
 145 143 133 127 131 136 126 141 129 124 141 134 136 138 138 136 134 139
 144 134 128 141 134 127 138 128 132 141 145 136 125 129 130 133 129 126
 131 128 139 143 134 138 135 131 131 152 138 136 131 150 133 127 125 129
 137 135 140 132 137 147 146 137 131 138 126 122 128 140 130 124 136 141
 140 128 140 129 129 127 131 138 134 137 133 141 138 129 134 137 141 127
 128 116 136 136 132 136 139 147 122 128 124 132 135 130 138 133 142 133
 131 130 131 122 135 137 133 136 132 123 133 134 140 121 139 140 123 137
 129 145 128 137 120 138 139 137 132 126 144 129 136 136 136 135 137 139
 129 132 129 135 143 128 138 135 140 141 121 135 145 146 136 140 132 122
 133 134 128 142 133 135]
```

Counts in ADU are quite small, not a lot of read out noise coming from the camera (compared to C14). From checking the first image, we can see that the counts are pretty much consistent.

Minimum, Maximum, Mean pixel counts in ADU as well as the Standard deviation

[4]:
```
#Prints the min(ADU), max(ADU), mean(ADU) and standard deviation for the counts␣
↪in the first bias image
print('Min:', np.min(biases[0]))
print('Max:', np.max(biases[0]))
print('Mean:', np.mean(biases[0]))
print('Standard Deviation:', np.std(biases[0]))
```

```
Min: 80
Max: 554
Mean: 112.72491862104319
Standard Deviation: 9.21282954487407
```

Standard

[5]:
```
# displaying the first bias image
fig, ax = plt.subplots(figsize = (12,7))
plt.rcParams.update({'font.size':14 })

plt.imshow(biases[0], cmap='gray', norm=LogNorm(), vmin = 90, vmax = 140)

plt.xlabel('x-pixels')
plt.ylabel('y-pixels')

plt.title('First Bias Image')
plt.colorbar()
```

[5]:
```
<matplotlib.colorbar.Colorbar at 0x7f1548ffe810>
```

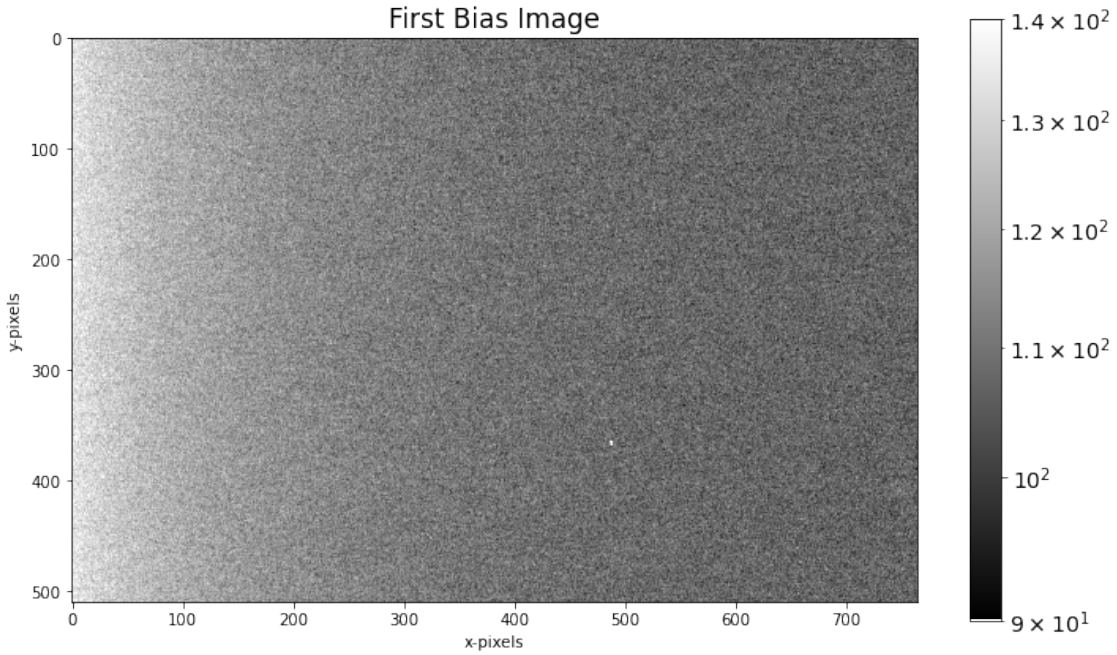

Figure 1. First bias image. The image for the bias is expected, however, there is a bright pixel spot

This appears on all the bias frames being used... This is going to subtract off from the dark, flat and science frames and therefore will affect the counts in that spot. Also notice the brightened left edge as mentioned earlier.

2.0.3 ⋆ Combining the Bias frames

Checking the stats for the combined bias frame

[6]:
```
# combines all bias frames together
bias_median = ccdproc.Combiner(biases).median_combine()

# prints bias median counts
print(bias_median)

# prints count statistics for the bias median
print('Min:', np.min(bias_median))
print('Max:', np.max(bias_median))
print('Mean:', np.mean(bias_median))
print('Standard Deviation:', np.std(bias_median))

# displays the image for the Bias Median
fig, ax = plt.subplots(figsize = (12,7))

plt.rcParams.update({'font.size':14 })
plt.imshow(bias_median, cmap='gray', norm=LogNorm(), vmax=140, vmin
=90)
plt.xlabel('x-pixels')
plt.ylabel('y-pixels')
plt.title('Bias Median')
plt.colorbar()
```

```
[[135.  132.5 133.5 … 103.5 103.5 103.5]
 [130.5 133.  132.5 … 105.  104.  103.5]
 [132.5 134.  135.5 … 105.  103.5 102. ]
 …
 [126.5 130.  130.  … 105.  104.5 104. ]
 [132.  131.5 129.5 … 105.  101.  106.5]
 [131.  130.  129.  … 105.  105.  104.5]]
Min: 97.0
Max: 542.5
Mean: 109.27966551326413
Standard Deviation: 6.784267733107446
```

[6]: <matplotlib.colorbar.Colorbar at 0x7f1548dbdfd0>

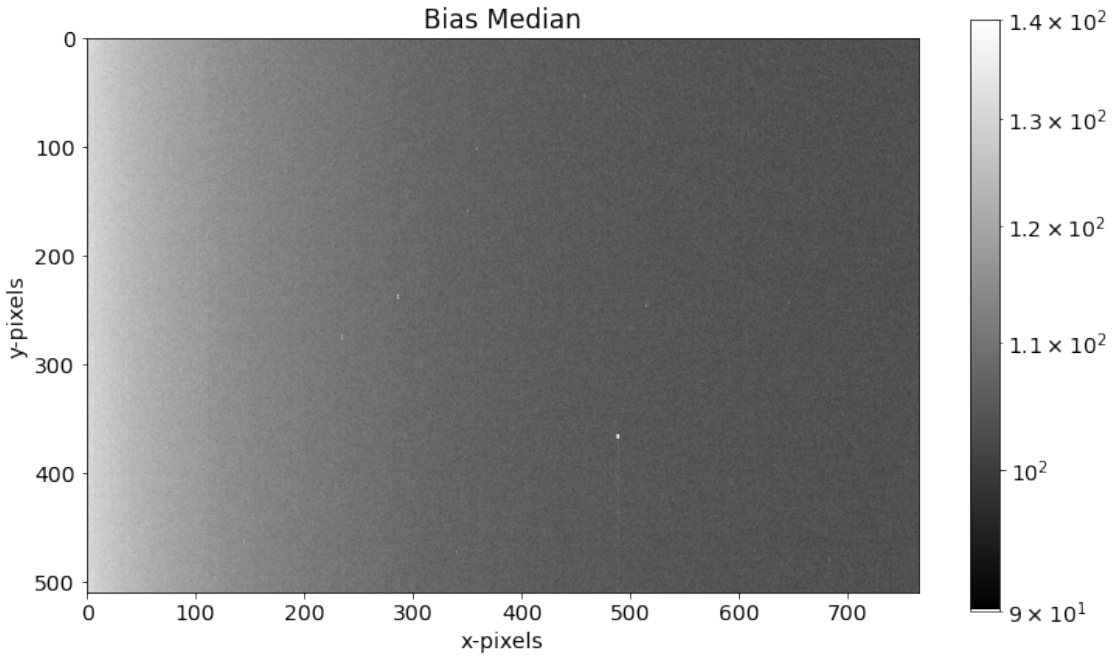

Figure 2. Bias Median. All biases have been combined. The bright pixel spot still remains. Overall, by combining the frames, the standard deviation decreases when compared to one single bias frame (as shown in **Figure 1**).

2.0.4 ⋆ Creating Bias median frame

[7]:
```
# header for bias median frame
bias_median.meta.update(EXPTIME = 0) # "add" exposure time "to header"
bias_median.meta.update(TELESCOP = 'C11') # "" telescope ""
bias_median.meta.update(OBJECT = 'Bias_Median') # "" image(object) type ""
print(bias_median.meta)

# creating bias_median.fits file
bias_median.write("bias_median.fits")

# removes the "biases" array
del(biases)
```

```
OrderedDict([('NCOMBINE', 28), ('EXPTIME', 0), ('TELESCOP', 'C11'), ('OBJECT',
'Bias_Median')])
```

3 Dark

3.0.1 ⋆ Collecting and Loading the Dark frames

[8]:
```
# creating image list
images = ccdproc.ImageFileCollection(".",glob_include = 'Dark_600*')

print("List of Dark Frames")
for fn in images.files_filtered(PICTTYPE = 3):
    print(fn)

# filter darks, reading the dark frames in ADU and defining them into list␣
 ↪'darks'
darks = [CCDData.read(fn, unit = "adu") for fn in images.
 ↪files_filtered(PICTTYPE = 3)]
```

```
List of Dark Frames
Dark_600.000secs_-5.23C_00002340.fit
Dark_600.000secs_-5.23C_00002341.fit
Dark_600.000secs_-5.23C_00002342.fit
Dark_600.000secs_-5.23C_00002345.fit
Dark_600.000secs_-5.23C_00002346.fit
Dark_600.000secs_-5.23C_00002347.fit
Dark_600.000secs_-5.23C_00002349.fit
Dark_600.000secs_-5.23C_00002351.fit
Dark_600.000secs_-5.23C_00002353.fit
Dark_600.000secs_-5.23C_00002354.fit
Dark_600.000secs_-5.23C_00002355.fit
Dark_600.000secs_-5.23C_00002359.fit
Dark_600.000secs_-5.23C_00002360.fit
Dark_600.000secs_-5.23C_00002365.fit
Dark_600.000secs_-5.23C_00002367.fit
Dark_600.000secs_-5.23C_00002368.fit
Dark_600.000secs_-5.23C_00002369.fit
Dark_600.000secs_-5.23C_00002371.fit
Dark_600.000secs_-5.23C_00002377.fit
Dark_600.000secs_-5.23C_00002380.fit
```

3.0.2 ⋆ Checking individual Dark Frames

Checking each dark frame in DS9 In terminal: “ds9 -zscale Dark_* &” - First check is to identify if they are all dark frames. This is true - All frames appear to have more light coming in from the left side of the images (this occured with the bias images as well), dark frames are useable

NOTE the Temperature! Science images and darks should have the same temperature within 0.5 degrees - all Darks have the same temperature and exposure time

[9]:
```
# displaying exposure time and temperature of the first dark frame
print("Exposure time of Darks: ", darks[0].header['EXPTIME'], "\n", "  ␣
 ↪Temperature of CCD: ", darks[0].header['CCD-TEMP'])
```

```
Exposure time of Darks:  600.0
    Temperature of CCD:  -5.23215684598954
```

Checking the statisitics for the first frame

[10]:
```
print("Pixel Value (ADU): All Dark Images", "\n", darks, "\n") # prints list of␣
 ↪pixel values for all the dark frames
print("Pixel Value (ADU): First Dark Image", "\n", darks[0], "\n") # prints the␣
 ↪pixel values of the first dark frames
print("Pixel Value (ADU): First Dark Image, First Column:", "\n", darks[0][:
 ↪,0], "\n") # print the counts of the first column of the first dark frames

# stats for first frame
print('Min:', np.min(darks[0]))
print('Max:', np.max(darks[0]))
print('Mean:', np.mean(darks[0]))
print('Standard Deviation:', np.std(darks[0]))

#Displays the image for the first dark image
fig, ax = plt.subplots(figsize = (12,7))

plt.rcParams.update({'font.size':14 })
plt.imshow(darks[1], cmap='gray', norm=LogNorm(), vmin = 95, vmax =172)
plt.xlabel('x-pixels')
plt.ylabel('y-pixels')
plt.title('First Raw Dark Image')
plt.colorbar()
```

```
Pixel Value (ADU): All Dark Images
 [CCDData([[145, 140, 145, …, 111, 111, 120],
          [150, 141, 151, …, 119, 117, 156],
          [132, 134, 153, …, 117, 112, 124],
          …,
          [186, 133, 146, …, 113, 121, 114],
          [134, 139, 141, …, 112, 117, 125],
          [154, 143, 143, …, 119, 116, 118]]), CCDData([[133, 141, 135, …,
119, 117, 109],
          [132, 143, 147, …, 112, 109, 151],
          [138, 145, 137, …, 112,  99, 101],
          …,
          [188, 138, 129, …, 130, 107, 104],
          [146, 143, 141, …, 121, 109, 118],
          [122, 139, 132, …, 113, 119, 109]]), CCDData([[130, 130, 132, …,
```

```
97, 112, 111],
         [129, 124, 121, …, 100, 107, 134],
         [122, 130, 133, …,  94,  98, 106],
         …,
         [170, 128, 144, …, 111, 102, 109],
         [133, 145, 118, …, 110, 115, 105],
         [140, 137, 125, …, 114, 113, 112]]), CCDData([[138, 135, 142, …,
109,  98, 102],
         [135, 132, 126, …, 106, 101, 135],
         [142, 118, 132, …, 103, 113, 103],
         …,
         [172, 132, 133, …, 109, 108, 107],
         [140, 127, 138, …, 109, 120, 100],
         [130, 136, 133, …,  99,  97, 111]]), CCDData([[135, 142, 130, …,
90, 105, 114],
         [138, 123, 123, …, 108, 109, 143],
         [134, 122, 124, …, 100, 107, 103],
         …,
         [175, 134, 131, …, 105, 120, 100],
         [137, 122, 138, …, 109, 113, 115],
         [127, 142, 125, …, 111, 104, 115]]), CCDData([[135, 149, 141, …,
119, 110, 113],
         [147, 135, 142, …, 101, 108, 155],
         [137, 148, 134, …, 105, 121, 103],
         …,
         [192, 144, 145, …, 111, 114, 112],
         [143, 141, 138, …, 117, 118, 120],
         [147, 151, 140, …, 105, 112, 121]]), CCDData([[140, 136, 135, …,
110, 109, 127],
         [142, 127, 136, …, 114, 106, 141],
         [136, 138, 138, …, 102, 113,  99],
         …,
         [177, 138, 145, …, 112, 114, 118],
         [149, 131, 137, …, 115, 116, 112],
         [143, 144, 139, …, 117, 118, 115]]), CCDData([[143, 135, 143, …,
108, 109, 112],
         [136, 130, 132, …, 111, 107, 147],
         [126, 136, 134, …,  96, 102, 105],
         …,
         [187, 137, 131, …, 118, 108, 114],
         [133, 139, 138, …, 110, 107, 106],
         [148, 134, 140, …, 104, 113, 100]]), CCDData([[135, 118, 130, …,
116, 116, 110],
         [140, 127, 126, …, 110, 102, 142],
         [124, 138, 133, …, 104, 109, 100],
         …,
         [167, 151, 133, …, 107, 110, 109],
         [125, 131, 130, …, 109, 112, 104],
```

```
         [125, 127, 135, …, 107, 101, 121]]), CCDData([[132, 137, 150, …,
109, 106, 117],
         [135, 144, 138, …, 108, 111, 151],
         [135, 133, 132, …, 107, 107, 107],
         …,
         [186, 142, 137, …, 107, 111, 109],
         [150, 146, 147, …, 113, 124, 110],
         [138, 136, 144, …, 115, 108, 111]]), CCDData([[148, 137, 131, …,
106, 101, 117],
         [129, 143, 122, …, 104, 113, 156],
         [144, 137, 137, …, 121, 103, 107],
         …,
         [192, 134, 119, …, 118, 106, 109],
         [137, 136, 135, …, 109, 104, 122],
         [137, 143, 127, …, 103,  98, 114]]), CCDData([[139, 138, 134, …,
102, 108, 112],
         [134, 138, 133, …, 102, 110, 136],
         [133, 132, 142, …, 111,  98, 108],
         …,
         [179, 120, 145, …, 112, 106, 107],
         [133, 119, 134, …, 107, 110, 120],
         [135, 133, 134, …, 109, 115, 118]]), CCDData([[126, 140, 144, …,
118, 110, 117],
         [122, 135, 133, …, 114, 113, 150],
         [140, 135, 135, …, 111, 116, 113],
         …,
         [186, 138, 140, …, 106, 116, 119],
         [139, 136, 141, …, 124,  99, 113],
         [137, 157, 128, …, 109, 104, 112]]), CCDData([[139, 135, 127, …,
105, 107, 111],
         [130, 141, 117, …, 102, 105, 137],
         [143, 143, 121, …, 100, 107, 103],
         …,
         [170, 132, 133, …, 106, 112, 104],
         [140, 134, 137, …, 117, 113, 100],
         [130, 138, 134, …, 116, 109, 117]]), CCDData([[128, 140, 141, …,
105, 101, 109],
         [133, 131, 138, …, 103, 117, 141],
         [128, 140, 139, …, 110, 107, 104],
         …,
         [179, 132, 133, …, 112, 112, 110],
         [130, 151, 140, …, 109, 104, 109],
         [133, 132, 144, …, 106, 103, 119]]), CCDData([[128, 137, 138, …,
105,  99, 107],
         [142, 133, 126, …, 115, 108, 137],
         [142, 130, 141, …, 108, 110, 103],
         …,
         [179, 134, 137, …, 105, 106, 116],
```

```
         [148, 129, 130, …, 116, 114, 112],
         [137, 127, 134, …, 116, 116, 108]]), CCDData([[144, 153, 146, …,
111, 115, 113],
         [159, 132, 136, …, 115, 121, 143],
         [140, 144, 132, …, 118, 104, 111],
         …,
         [185, 146, 144, …, 111, 121, 121],
         [152, 149, 136, …, 119, 122, 108],
         [142, 153, 142, …, 114, 112, 120]]), CCDData([[132, 139, 131, …,
113, 102, 111],
         [138, 133, 143, …, 104,  98, 154],
         [141, 146, 134, …, 108, 104, 110],
         …,
         [170, 143, 136, …, 109, 117, 123],
         [143, 134, 137, …, 119,  99, 116],
         [140, 148, 129, …, 110, 111, 110]]), CCDData([[126, 139, 139, …,
97, 113, 108],
         [138, 152, 132, …, 108, 107, 137],
         [132, 131, 140, …, 114, 109, 105],
         …,
         [187, 134, 137, …, 115, 110, 106],
         [152, 145, 139, …, 108, 104, 107],
         [129, 136, 128, …, 113,  99, 103]]), CCDData([[140, 153, 146, …,
109, 108, 112],
         [125, 138, 146, …, 102, 105, 150],
         [129, 158, 139, …,  94, 105, 107],
         …,
         [188, 135, 141, …, 104, 120, 117],
         [144, 145, 139, …, 111, 108, 108],
         [145, 141, 143, …, 126, 110, 103]])]

Pixel Value (ADU): First Dark Image
 [[145 140 145 … 111 111 120]
 [150 141 151 … 119 117 156]
 [132 134 153 … 117 112 124]
 …
 [186 133 146 … 113 121 114]
 [134 139 141 … 112 117 125]
 [154 143 143 … 119 116 118]]

Pixel Value (ADU): First Dark Image, First Column:
 [ 145  150  132  143  144 1019  150  138  132  137  136  140  138  139
  146  135  145  128  142  135  129  140  129  141  148  195  137  146
  133  145  144  133  144  152  142  145  137  142  138  134  147  141
  139  136  135  127  138  139  136  138  142  149  143  130  143  136
  138  137  146  145  140  144  146  144  140  134  146  140  139  144
  142  197  151  147  134  137  148  141  139  134  144  138  149  190
  137  142  139  145  141  152  143  141  139  136  142  148  147  143
```

```
 145  142  144  167  130  135  148  145  137  136  142  154  140  141
 132  145  140  145  152  133  137  143  156  130  146  150  133  140
 144  149  140  145  148  139  137  149  144  150  139  134  140  151
 155  143  145  135  141  148  140  138  140  132  150  134  138  137
 146  152  138  165  139  135  141  146  139  133  149  143  137  146
 141  136  145  148  140  140  139  136  140  135  149  131  143  140
 159  146  144  144  143  139  138  138  135  138  148  147  139  159
 138  141  137  136  144  143  137  147  137  142  150  132  141  144
 197  137  151  144  144  142  148  148  148  141  135  141  145  154
 142  143  142  143  148  146  142  127  136  151  142  138  134  140
 140  144  139  147  141  147  141  145  136  137  141  147  146  139
 143  134  124  147  150  147  146  139  146  146  150  146  129  139
 143  143  154  153  144  140  145  143  129  149  140  142  135  151
 139  140  143  143  127  139  151  131  138  144  144  142  133  145
 144  151  137  138  140  145  143  132  129  132  173  148  144  148
 137  151  151  144  139  150  146  143  137  145  140  152  139  132
 133  130  144  140  139  148  144  134  132  139  141  134  136  138
 147  144  141  139  151  141  140  157  152  150  137  138  144  150
 151  146  140  144  152  144  134  138  137  150  129  136  143  140
 134  152  142  180  142  145  147  145  155  150  152  140  140  142
 148  140  140  138  140  145  148  140  140  139  133  142  144  155
 150  139  892  142  144  152  140  148  148  152  138  142  134  131
 144  138  140  145  136  153  143  146  127  150  133  143  135  151
 135  139  133  138  144  135  140  134  148  137  137  144  142  131
 148  136  153  146  138  139  144  145  131  134  132  134  136  129
 137  139  145  136  139  134  153  145  148  128  146  144  139  148
 139  134  141  140  129  140  143  148  154  137  147  153  145  150
 140  128  155  140  146  142  140  135  151  141  142  145  137  146
 144  132  147  141  144  141  133  150  133  152  156  144  145  147
 197  140  155  186  134  154]

Min: 83
Max: 54958
Mean: 125.62277841855696
Standard Deviation: 215.13966165753033
```

[10]: <matplotlib.colorbar.Colorbar at 0x7f157c625fd0>

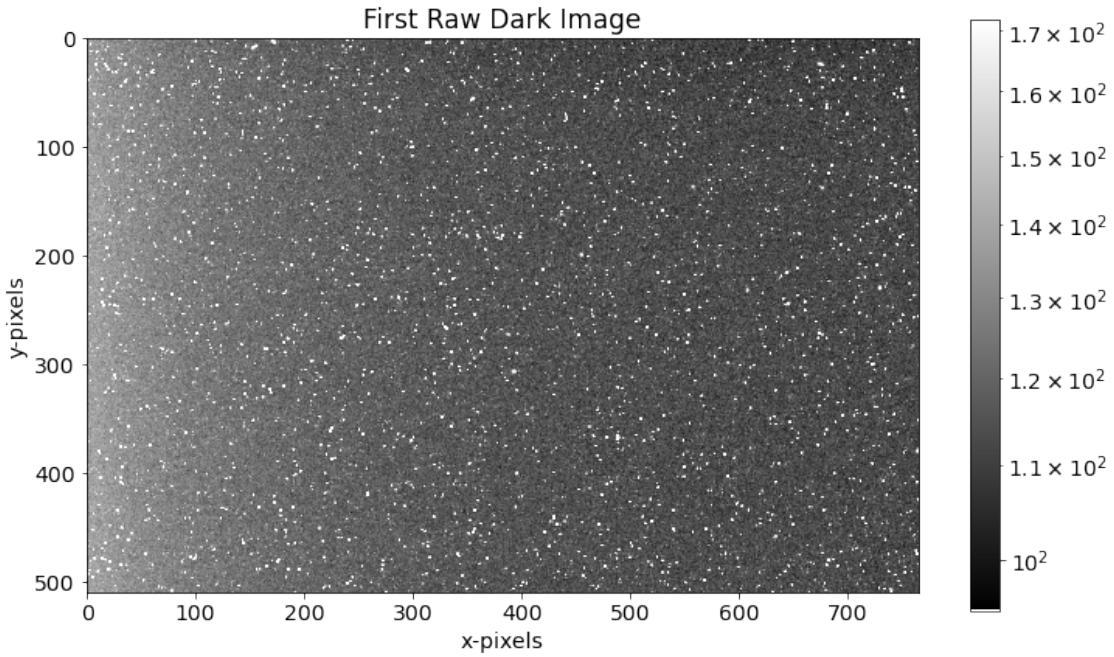

Figure 3. First Raw Dark Image. Minimum value of counts is 83 adu and max is 54958 adu, varius hot pixels can be seen scattered about in the image, this therefore will give a large standard deviation. In this case our std is 215 which is high. This likely means there was a lot of random noise when the exposure was taken.

3.0.3 ⋆ Subtracting the Bias from all the indiviual Dark exposures

[11]:
```
# for loop which goes through each of the "darks" images and subtracts the␣
 ↪value of the "bias_median"
for idx, thisimage in enumerate(darks):
    darks[idx] = ccdproc.subtract_bias(thisimage, bias_median)
```

Checking the statisitics for the first bias subtracted dark frame

[12]:
```
# Prints out the list (array) of pixel values for processed dark
print("Pixel Value (ADU): All Dark Images", "\n", darks[0], "\n")


#Prints the statistic counts for first processed dark frame (min, max, mean and␣
 ↪standard deviation)
print('Min:', np.min(darks[0]))
print('Max:', np.max(darks[0]))
print('Mean:', np.mean(darks[0]))
print('Standard Deviation:', np.std(darks[0]))
```

```
#Displays the image for the first dark frame
fig, ax = plt.subplots(figsize = (12,7))

plt.rcParams.update({'font.size':14 })
plt.imshow(darks[0], cmap='gray', norm=LogNorm(),vmin = 95, vmax =172)
plt.xlabel('x-pixels')
plt.ylabel('y-pixels')
plt.title('First Processed Dark Image')
plt.colorbar()
```

```
Pixel Value (ADU): All Dark Images
 [[10.   7.5 11.5 … 7.5  7.5 16.5]
 [19.5  8.  18.5 … 14.  13.  52.5]
 [-0.5  0.  17.5 … 12.   8.5 22. ]
 …
 [59.5  3.  16.  … 8.  16.5 10. ]
 [ 2.   7.5 11.5 … 7.  16.  18.5]
 [23.  13.  14.  … 14.  11.  13.5]]

Min: -88.0
Max: 54415.5
Mean: 16.343112905292838
Standard Deviation: 214.62105473397972
```

[12]: <matplotlib.colorbar.Colorbar at 0x7f154840b110>

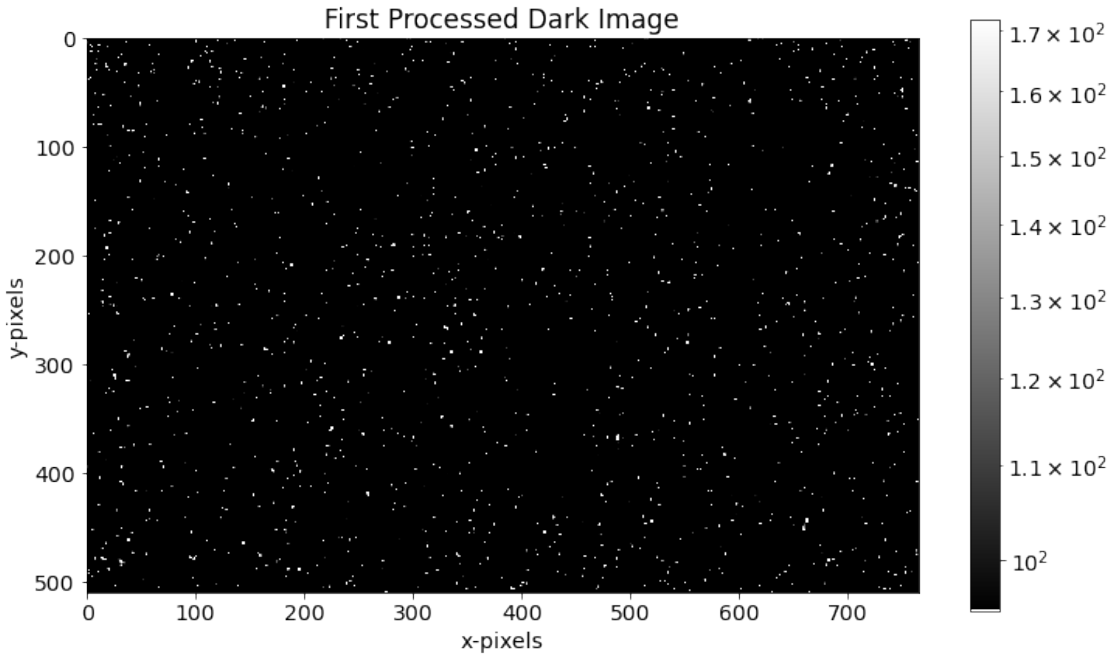

Figure 4. By subtracting the Bias Median from the First Dark, we can see a decrease in pixel count.

- Mean count was ~120 ADU, this dropped to ~16 ADU after bias median count was subtracted off (a decrease in ~ 109 ADU). Hence why the pixels (image) are a lot darker in figure 4 when compared to figure 3.

Checking if it was successful by printing the stats for all the processed darks

[13]:
```
print("Pixel Value (ADU): All Processed Dark Images", "\n", darks, "\n") #␣
 ↪prints list of pixel values for all the processed dark frames
```

```
Pixel Value (ADU): All Processed Dark Images
 [CCDData([[10. ,  7.5, 11.5, …,  7.5,  7.5, 16.5],
         [19.5,  8. , 18.5, …, 14. , 13. , 52.5],
         [-0.5,  0. , 17.5, …, 12. ,  8.5, 22. ],
         …,
         [59.5,  3. , 16. , …,  8. , 16.5, 10. ],
         [ 2. ,  7.5, 11.5, …,  7. , 16. , 18.5],
         [23. , 13. , 14. , …, 14. , 11. , 13.5]]), CCDData([[-2. ,  8.5,
1.5, …, 15.5, 13.5,  5.5],
         [ 1.5, 10. , 14.5, …,  7. ,  5. , 47.5],
         [ 5.5, 11. ,  1.5, …,  7. , -4.5, -1. ],
         …,
         [61.5,  8. , -1. , …, 25. ,  2.5,  0. ],
         [14. , 11.5, 11.5, …, 16. ,  8. , 11.5],
         [-9. ,  9. ,  3. , …,  8. , 14. ,  4.5]]), CCDData([[ -5. ,  -2.5,
-1.5, …,  -6.5,   8.5,   7.5],
         [ -1.5,  -9. , -11.5, …,  -5. ,   3. ,  30.5],
         [-10.5,  -4. ,  -2.5, …, -11. ,  -5.5,   4. ],
         …,
         [ 43.5,  -2. ,  14. , …,   6. ,  -2.5,   5. ],
         [  1. ,  13.5, -11.5, …,   5. ,  14. ,  -1.5],
         [  9. ,   7. ,  -4. , …,   9. ,   8. ,   7.5]]), CCDData([[  3. ,
2.5,   8.5, …,   5.5,  -5.5,  -1.5],
         [  4.5,  -1. ,  -6.5, …,   1. ,  -3. ,  31.5],
         [  9.5, -16. ,  -3.5, …,  -2. ,   9.5,   1. ],
         …,
         [ 45.5,   2. ,   3. , …,   4. ,   3.5,   3. ],
         [  8. ,  -4.5,   8.5, …,   4. ,  19. ,  -6.5],
         [ -1. ,   6. ,   4. , …,  -6. ,  -8. ,   6.5]]), CCDData([[  0. ,
9.5,  -3.5, …, -13.5,   1.5,  10.5],
         [  7.5, -10. ,  -9.5, …,   3. ,   5. ,  39.5],
         [  1.5, -12. , -11.5, …,  -5. ,   3.5,   1. ],
         …,
         [ 48.5,   4. ,   1. , …,   0. ,  15.5,  -4. ],
         [  5. ,  -9.5,   8.5, …,   4. ,  12. ,   8.5],
         [ -4. ,  12. ,  -4. , …,   6. ,  -1. ,  10.5]]), CCDData([[ 0. ,
```

```
16.5,  7.5, …, 15.5,  6.5,  9.5],
        [16.5,  2. ,  9.5, …, -4. ,  4. , 51.5],
        [ 4.5, 14. , -1.5, …,  0. , 17.5,  1. ],
        …,
        [65.5, 14. , 15. , …,  6. ,  9.5,  8. ],
        [11. ,  9.5,  8.5, …, 12. , 17. , 13.5],
        [16. , 21. , 11. , …,  0. ,  7. , 16.5]]), CCDData([[ 5. ,  3.5,
1.5, …,  6.5,  5.5, 23.5],
        [11.5, -6. ,  3.5, …,  9. ,  2. , 37.5],
        [ 3.5,  4. ,  2.5, …, -3. ,  9.5, -3. ],
        …,
        [50.5,  8. , 15. , …,  7. ,  9.5, 14. ],
        [17. , -0.5,  7.5, …, 10. , 15. ,  5.5],
        [12. , 14. , 10. , …, 12. , 13. , 10.5]]), CCDData([[ 8. ,  2.5,
9.5, …,  4.5,  5.5,  8.5],
        [ 5.5, -3. , -0.5, …,  6. ,  3. , 43.5],
        [-6.5,  2. , -1.5, …, -9. , -1.5,  3. ],
        …,
        [60.5,  7. ,  1. , …, 13. ,  3.5, 10. ],
        [ 1. ,  7.5,  8.5, …,  5. ,  6. , -0.5],
        [17. ,  4. , 11. , …, -1. ,  8. , -4.5]]), CCDData([[  0. , -14.5,
-3.5, …,  12.5,  12.5,   6.5],
        [  9.5,  -6. ,  -6.5, …,   5. ,  -2. ,  38.5],
        [ -8.5,   4. ,  -2.5, …,  -1. ,   5.5,  -2. ],
        …,
        [ 40.5,  21. ,   3. , …,   2. ,   5.5,   5. ],
        [ -7. ,  -0.5,   0.5, …,   4. ,  11. ,  -2.5],
        [ -6. ,  -3. ,   6. , …,   2. ,  -4. ,  16.5]]), CCDData([[-3. ,
4.5, 16.5, …,  5.5,  2.5, 13.5],
        [ 4.5, 11. ,  5.5, …,  3. ,  7. , 47.5],
        [ 2.5, -1. , -3.5, …,  2. ,  3.5,  5. ],
        …,
        [59.5, 12. ,  7. , …,  2. ,  6.5,  5. ],
        [18. , 14.5, 17.5, …,  8. , 23. ,  3.5],
        [ 7. ,  6. , 15. , …, 10. ,  3. ,  6.5]]), CCDData([[ 13. ,   4.5,
-2.5, …,   2.5,  -2.5,  13.5],
        [ -1.5,  10. , -10.5, …,  -1. ,   9. ,  52.5],
        [ 11.5,   3. ,   1.5, …,  16. ,  -0.5,   5. ],
        …,
        [ 65.5,   4. , -11. , …,  13. ,   1.5,   5. ],
        [  5. ,   4.5,   5.5, …,   4. ,   3. ,  15.5],
        [  6. ,  13. ,  -2. , …,  -2. ,  -7. ,   9.5]]), CCDData([[  4. ,
5.5,   0.5, …,  -1.5,   4.5,   8.5],
        [  3.5,   5. ,   0.5, …,  -3. ,   6. ,  32.5],
        [  0.5,  -2. ,   6.5, …,   6. ,  -5.5,   6. ],
        …,
        [ 52.5, -10. ,  15. , …,   7. ,   1.5,   3. ],
        [  1. , -12.5,   4.5, …,   2. ,   9. ,  13.5],
```

```
        [  4. ,   3. ,   5. , …,   4. ,  10. ,  13.5]]), CCDData([[-9. ,
7.5, 10.5, …, 14.5,  6.5, 13.5],
        [-8.5,  2. ,  0.5, …,  9. ,  9. , 46.5],
        [ 7.5,  1. , -0.5, …,  6. , 12.5, 11. ],
        …,
        [59.5,  8. , 10. , …,  1. , 11.5, 15. ],
        [ 7. ,  4.5, 11.5, …, 19. , -2. ,  6.5],
        [ 6. , 27. , -1. , …,  4. , -1. ,  7.5]]), CCDData([[  4. ,   2.5,
-6.5, …,   1.5,   3.5,   7.5],
        [ -0.5,   8. , -15.5, …,  -3. ,   1. ,  33.5],
        [ 10.5,   9. , -14.5, …,  -5. ,   3.5,   1. ],
        …,
        [ 43.5,   2. ,   3. , …,   1. ,   7.5,   0. ],
        [  8. ,   2.5,   7.5, …,  12. ,  12. ,  -6.5],
        [ -1. ,   8. ,   5. , …,  11. ,   4. ,  12.5]]), CCDData([[-7. ,
7.5,  7.5, …,  1.5, -2.5,  5.5],
        [ 2.5, -2. ,  5.5, …, -2. , 13. , 37.5],
        [-4.5,  6. ,  3.5, …,  5. ,  3.5,  2. ],
        …,
        [52.5,  2. ,  3. , …,  7. ,  7.5,  6. ],
        [-2. , 19.5, 10.5, …,  4. ,  3. ,  2.5],
        [ 2. ,  2. , 15. , …,  1. , -2. , 14.5]]), CCDData([[-7. ,  4.5,
4.5, …,  1.5, -4.5,  3.5],
        [11.5,  0. , -6.5, …, 10. ,  4. , 33.5],
        [ 9.5, -4. ,  5.5, …,  3. ,  6.5,  1. ],
        …,
        [52.5,  4. ,  7. , …,  0. ,  1.5, 12. ],
        [16. , -2.5,  0.5, …, 11. , 13. ,  5.5],
        [ 6. , -3. ,  5. , …, 11. , 11. ,  3.5]]), CCDData([[ 9. , 20.5,
12.5, …,  7.5, 11.5,  9.5],
        [28.5, -1. ,  3.5, …, 10. , 17. , 39.5],
        [ 7.5, 10. , -3.5, …, 13. ,  0.5,  9. ],
        …,
        [58.5, 16. , 14. , …,  6. , 16.5, 17. ],
        [20. , 17.5,  6.5, …, 14. , 21. ,  1.5],
        [11. , 23. , 13. , …,  9. ,  7. , 15.5]]), CCDData([[-3. ,  6.5,
-2.5, …,  9.5, -1.5,  7.5],
        [ 7.5,  0. , 10.5, …, -1. , -6. , 50.5],
        [ 8.5, 12. , -1.5, …,  3. ,  0.5,  8. ],
        …,
        [43.5, 13. ,  6. , …,  4. , 12.5, 19. ],
        [11. ,  2.5,  7.5, …, 14. , -2. ,  9.5],
        [ 9. , 18. ,  0. , …,  5. ,  6. ,  5.5]]), CCDData([[-9. ,  6.5,
5.5, …, -6.5,  9.5,  4.5],
        [ 7.5, 19. , -0.5, …,  3. ,  3. , 33.5],
        [-0.5, -3. ,  4.5, …,  9. ,  5.5,  3. ],
        …,
        [60.5,  4. ,  7. , …, 10. ,  5.5,  2. ],
```

```
        [20. , 13.5,  9.5, …,  3. ,  3. ,  0.5],
        [-2. ,  6. , -1. , …,  8. , -6. , -1.5]]), CCDData([[  5. ,  20.5,
12.5, …,   5.5,   4.5,   8.5],
        [ -5.5,   5. ,  13.5, …,  -3. ,   1. ,  46.5],
        [ -3.5,  24. ,   3.5, …, -11. ,   1.5,   5. ],
        …,
        [ 61.5,   5. ,  11. , …,  -1. ,  15.5,  13. ],
        [ 12. ,  13.5,   9.5, …,   6. ,   7. ,   1.5],
        [ 14. ,  11. ,  14. , …,  21. ,   5. ,  -1.5]])]
```

Counts are all pretty much consistent with the first frame - Some are close to 0 counts as expected - The pixels with greater counts in ADU are most likely from the hot pixels and cosmic rays

3.0.4 ⋆ Combining the processed Dark exposures

[14]:
```
# combines the median dark value images together
dark_median = ccdproc.Combiner(darks).median_combine()
```

Checking stats

[15]:
```
# Prints out the list (array) of pixel values for the dark median
print("Pixel Value (ADU): Dark Median", "\n", dark_median, "\n")

#Prints the statistic counts for dark median (min, max, mean and standard␣
 ↪deviation)
print('Min:', np.min(dark_median))
print('Max:', np.max(dark_median))
print('Mean:', np.mean(dark_median))
print('Standard Deviation:', np.std(dark_median))

#Displays the image for the dark median image
fig, ax = plt.subplots(figsize = (12,7))

plt.rcParams.update({'font.size':14 })
plt.imshow(dark_median, cmap='gray', norm=LogNorm(), vmin = 95, vmax =172)
plt.xlabel('x-pixels')
plt.ylabel('y-pixels')
plt.title('Dark Median ')
plt.colorbar()
```

```
Pixel Value (ADU): Dark Median
 [[ 0.   6.   5.  …  5.5  5.   8.5]
 [ 5.   1.   0.5 …  3.   4.  39.5]
 [ 3.   2.5 -1.  …  2.5  3.5  3. ]
 …
 [55.5  4.5  7.  …  6.   7.   5.5]
```

```
 [ 8.   6.   8.5 …  6.5 11.5  4.5]
 [ 6.   8.5  5.  …  7.   5.5  8.5]]

Min: -77.0
Max: 54417.0
Mean: 12.463223119313085
Standard Deviation: 213.09233951134513
```

[15]: <matplotlib.colorbar.Colorbar at 0x7f1548327d50>

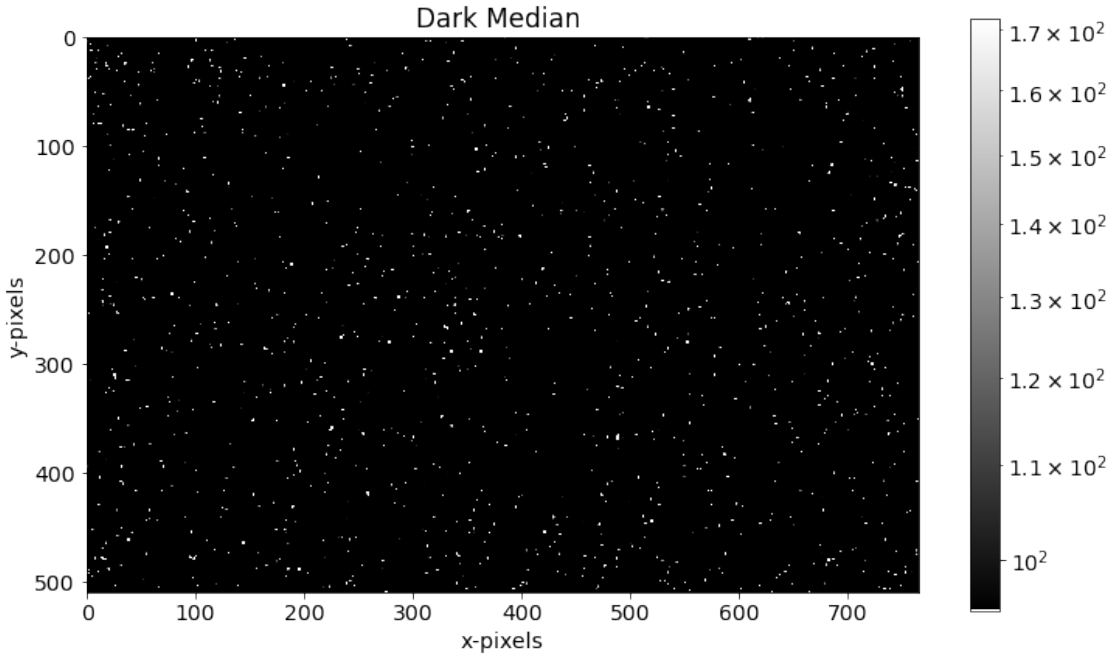

Figure 5. Dark Median. Final image that will subtract the outlying pixels (i.e. hot pixels) from the science images. The counts have fallen ~ 109 ADU.

- Standard deviation did not change much meaning that the added combined images roughly had noise in the same areas.

3.0.5 ⋆ Creating Dark median frame

[16]:
```
# header for the Dark Median frame
dark_median.meta.update(EXPTIME = 600) # exposure time
dark_median.meta.update(TELESCOP = 'C11') # telescope
dark_median.meta.update(OBJECT = 'Dark_Median') # image(object) type
print(dark_median.meta)

# creating dark_median.fits file
dark_median.write("dark_median.fits")
```

```
OrderedDict([('NCOMBINE', 20), ('EXPTIME', 600), ('TELESCOP', 'C11'), ('OBJECT',
'Dark_Median')])
```

[ ]: