Instructions

The following code was designed in order to implement batch processing of location tracking. Currently, for this to work, all videos must take the same settings, including the defined regions of interest. To confirm that regions of interest are consistent across videos, for each video an overlay of the reference frame, the regions of interest and the animal trace are produced. Moreover, the reference frame is currently required to be generated by taking the median of each individual video. In addition to saving frame by frame location, distance travelled, and whether the animal is in each ROI in separate csv files, bins can also be defined for summarizing videos (e.g. minute by minute). All summary information will be saved in a single file.

1. Load Necessary Packages

The following code loads neccessary packages and need not be changed by the user.

In [1]:
```
%load_ext autoreload
%autoreload 2
import os
import holoviews as hv
import numpy as np
import pandas as pd
import LocationTracking_Functions as lt
```

2. User Defines Batch Processing Directory and Settings

Below, options are set by user for batch processing of videos. If you are unfamiliar with these settings, please see LocationTracking_Individual.ipynb.

In [3]:
```
#define video paramaters
video_dict = {
    'dpath'        : r"G:\20211128-LPS-depression\oft\VideoOutput",
    'ftype'        : 'avi',
    'start'        : 0,
    'end'          : None,
    'region_names' : ['Area','outer','center'], #['Left','Right']
    'dmpl'         : 1,
    'stretch'      : dict(width=1, height=1)
}

#define parameters for location tracking
tracking_params = {
    'loc_thresh'    : 99,
    'use_window'    : True,
    'window_size'   : 100,
    'window_weight' : .9,
    'method'        : 'abs',
    'rmv_wire'      : False,
    'wire_krn'      : 10
}

#set bin_dict
#set bin_dict = None if only overall session average is desired
bin_dict = None


#code below loads folder with files.
video_dict = lt.Batch_LoadFiles(video_dict)
video_dict['FileNames']
```

Out[3]:
```
['LPS+Flu-1.avi',
 'LPS+Flu-2.avi',
 'LPS+Flu-3.avi',
 'LPS+Flu-4.avi',
 'LPS-1.avi',
 'LPS-2.avi',
 'LPS-3.avi',
 'LPS-4.avi',
 'saline-1.avi',
 'saline-2.avi',
 'saline-3.avi',
 'saline-4.avi']
```

#set bin_dict #set bin_dict = None if only overall session average is desired bin_dict = { '1' : (0,10), '2' : (10,20), '3' : (20,30) }

3. (Optional) Crop Image if Desired

To crop video frame, after running code below, select box selection tool below image (square with a plus sign). To start drawing region to be included in analyis, double click image. Double click again to finalize region. If you decide to change region, it is best to rerun this cell and subsequent steps. Note that this is done based upon first video in folder.

In [ ]:
```
%%output size=100

img_crp, video_dict = lt.LoadAndCrop(video_dict, cropmethod='Box', fstfile=True)
img_crp
```

4. (Optional) Mask Internal Regions

The following code is used to exclude internal portion of image from the field of view. After running cell below, draw regions to be excluded. To start drawing a region, double click on image. Single click to add a vertex. Double click to close polygon. If you mess up it's easiest to re-run cell.

In [ ]:
```
%%output size=100

img_mask, video_dict['mask'] = lt.Mask_select(video_dict, fstfile=True)
img_mask
```

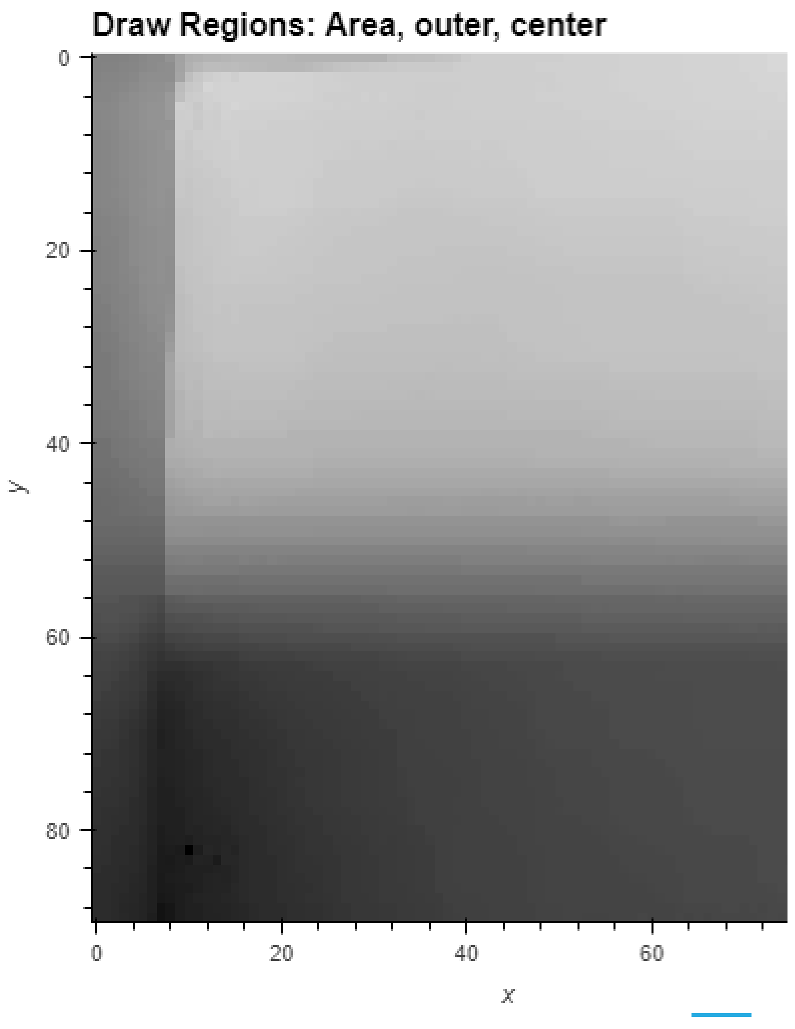
5. (Optional) Define Regions of Interest

After running cell below, draw regions of interest on presented image in the order you provided them. To start drawing a region, double click on image. Single click to add a vertex. Double click to close polygon. If you mess up it's easiest to re-run cell. Note that this is done based upon first video in folder.

In [4]:
```
%%output size=600

video_dict['reference'], img_ref = lt.Reference(video_dict, fstfile=True, num_frames=50)
img_roi, video_dict['roi_stream'] = lt.ROI_plot(video_dict)
img_roi
```

Out[4]:



6. (Optional) Define Scale for Distance Calculations

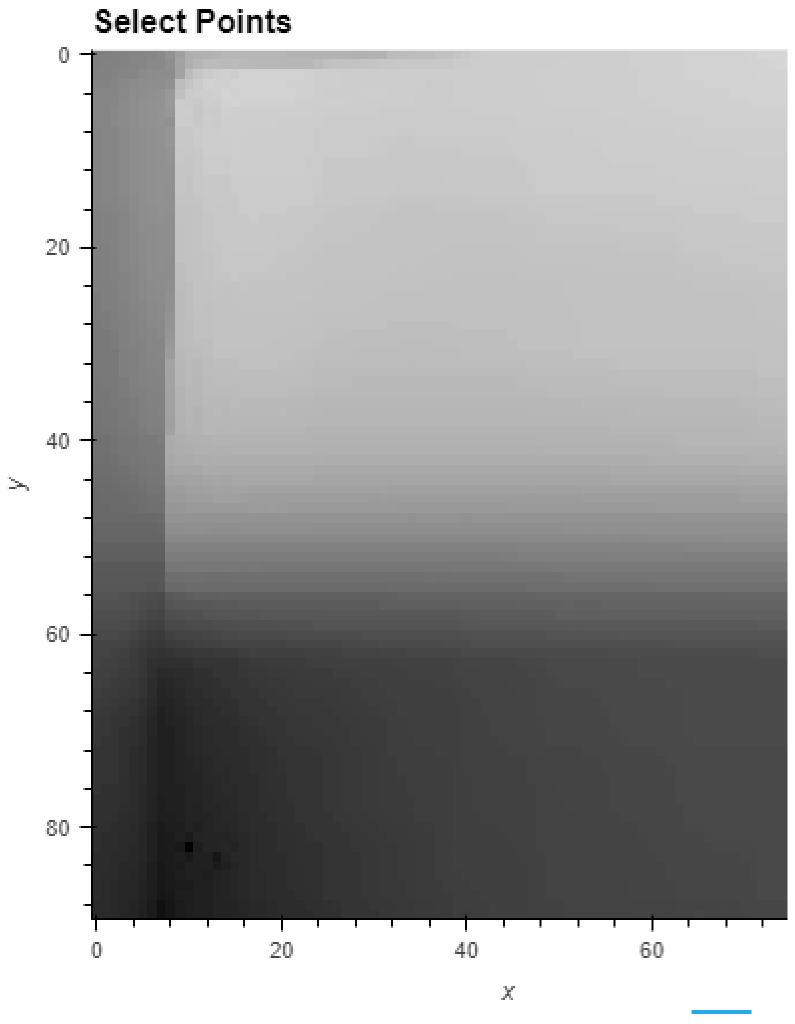
6a. Select two points of known distance

After running cell below, click on any two points and the distance between them, in pixel units, will be presented/returned. Will be used to convert pixel distance to other scale. Note that once drawn, points can be dragged or you can click again.

In [5]:
```
%%output size = 600

video_dict['reference'], img_ref = lt.Reference(video_dict, fstfile=True, num_frames=100)
img_scl, video_dict['scale'] = lt.DistanceTool(video_dict)
img_scl
```

Out[5]:



6b. Define real-world distance between points

Below, set the distance between the points selected, and the scale. Note that scale can be any desired text.

In [6]:
```
distance = 50
scale = 'cm'

video_dict['scale'] = lt.setScale(distance, scale, video_dict['scale'])
```

7. Perform Batch Processing and Display Traces from Each Session

The code below will save frame by frame data for each video file in its own csv. Binned summary information will be saved in a single file entitled 'BatchSummary.csv'. Additionally, the reference frame for each session will be displayed, along with a trace of the regions of interest (if supplied), and a trace of the animal's location across the session.

In [7]:
```
%%opts Layout [shared_axes=False]
%%output size=600

summary, images = lt.Batch_Process(video_dict,tracking_params,bin_dict)
images.cols(2)
```

```
Processing File: LPS+Flu-1.avi
total frames: 90093
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████
█████████████████████████████████| 90093/90093 [00:52<
00:00, 1706.40it/s]

total frames processed: 90093

Defining transitions...
Processing File: LPS+Flu-2.avi
total frames: 90093
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████
█████████████████████████████████| 90093/90093 [00:53<
00:00, 1680.90it/s]

total frames processed: 90093

Defining transitions...
Processing File: LPS+Flu-3.avi
total frames: 90093
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████
█████████████████████████████████| 90093/90093 [00:53<
00:00, 1697.86it/s]

total frames processed: 90093

Defining transitions...
Processing File: LPS+Flu-4.avi
total frames: 90093
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████
█████████████████████████████████| 90093/90093 [00:54<
00:00, 1651.26it/s]

total frames processed: 90093

Defining transitions...
Processing File: LPS-1.avi
total frames: 90092
nominal fps: 25.0
dimensions (h x w): 90,90
```

```
100%|█████████████████████████████████████████████████████████████████████████████████| 90092/90092 [00:54<00:00, 1641.33it/s]

total frames processed: 90092

Defining transitions...
Processing File: LPS-2.avi
total frames: 90092
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████████████████████████████████████████████| 90092/90092 [00:53<00:00, 1678.18it/s]

total frames processed: 90092

Defining transitions...
Processing File: LPS-3.avi
total frames: 90092
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████████████████████████████████████████████| 90092/90092 [00:52<00:00, 1713.35it/s]

total frames processed: 90092

Defining transitions...
Processing File: LPS-4.avi
total frames: 90092
nominal fps: 25.0
dimensions (h x w): 90,90

100%|█████████████████████████████████████████████████████████████████████████████████| 90092/90092 [00:54<00:00, 1646.28it/s]

total frames processed: 90092

Defining transitions...
Processing File: saline-1.avi
total frames: 0
nominal fps: 25.0
dimensions (h x w): 90,90

0it [00:00, ?it/s]

total frames processed: 0

Defining transitions...

---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
~\anaconda3\envs\ezTrack\lib\site-packages\pandas\core\indexes\range.py in get_loc(self, key, method, tolerance)
    384                 try:
```

```
--> 385                         return self._range.index(new_key)
    386                 except ValueError as err:

ValueError: 0 is not in range

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
C:\Users\ADMINI~1\AppData\Local\Temp/ipykernel_2832/3721950665.py in <module>
----> 1 summary, images = lt.Batch_Process(video_dict,tracking_params,bin_dict)
      2 images.cols(2)

~\ezTrack-1.2\LocationTracking\LocationTracking_Functions.py in Batch_Process(v
ideo_dict, tracking_params, bin_dict, accept_p_frames)
   1617
   1618         video_dict['reference'], image = Reference(video_dict,num_frames=50)
-> 1619         location = TrackLocation(video_dict,tracking_params)
   1620         location.to_csv(os.path.splitext(video_dict['fpath'])[0] + '_Locati
onOutput.csv', index=False)
   1621         file_summary = Summarize_Location(location, video_dict, bin_dict=bin_di
ct)

~\ezTrack-1.2\LocationTracking\LocationTracking_Functions.py in TrackLocation(v
ideo_dict, tracking_params)
    788         print('Defining transitions...')
    789         df['ROI_location'] = ROI_linearize(df[video_dict['region_names']])
--> 790         df['ROI_transition'] = ROI_transitions(df['ROI_location'])
    791
    792     #update scale, if known

~\ezTrack-1.2\LocationTracking\LocationTracking_Functions.py in ROI_transitions
(regions, include_first)
   1243
   1244     """
-> 1245     regions_offset = np.append(regions[0], regions[0:-1])
   1246     transitions = regions!=regions_offset
   1247     if include_first:

~\anaconda3\envs\ezTrack\lib\site-packages\pandas\core\series.py in __getitem_
_(self, key)
    940
    941         elif key_is_scalar:
--> 942             return self._get_value(key)
    943
    944         if is_hashable(key):

~\anaconda3\envs\ezTrack\lib\site-packages\pandas\core\series.py in _get_value
(self, label, takeable)
   1049
   1050         # Similar to Index.get_value, but we do not fall back to posit
ional
-> 1051         loc = self.index.get_loc(label)
   1052         return self.index._get_values_for_loc(self, loc, label)
   1053

~\anaconda3\envs\ezTrack\lib\site-packages\pandas\core\indexes\range.py in ge
t_loc(self, key, method, tolerance)
```

```
    385                     return self._range.index(new_key)
    386                 except ValueError as err:
--> 387                     raise KeyError(key) from err
    388             raise KeyError(key)
    389         return super().get_loc(key, method=method, tolerance=tolerance)

KeyError: 0
```