

1um Beads In Water at 3.33 fps (WFFM Hestia)

To run *and* export, paste the following command. Otherwise, export manually. You must first define monkier to do this.

```
% export("TrackMate_Live_Script_guide mlx",
strcat('C:\Users\al3xm\Documents\GitHub\HISTia\TrackMate Reports\',monkier),
HideCode=false, OpenExportedFile=true, Run=true, FigureResolution=1600)
```

Name this data analysis run

```
monkier = '1um beads in water at 3.33fps';
disp(monkier)
```

1um beads in water at 3.33fps

Set the current directory for data saving purposes.

```
cd('C:\Users\al3xm\Documents\GitHub\HISTia')
```

This Report generated at

```
disp(datetime)
```

20-Mar-2025 21:05:12

Define the source folder here:

```
main_folder = "C:\Users\al3xm\Documents\_Local_Data\25.03.20_1umBeadsWater\XMLs"
```

```
main_folder =
"C:\Users\al3xm\Documents\_Local_Data\25.03.20_1umBeadsWater\XMLs"
```

```
paths = FileFinder(main_folder, '.xml')
```

```
paths = 1x4 string
"C:\Users\al3xm\Documents\_Local_Data\25.03.20_1umBeadsWater\XMLs\tSeri... "C:\Us ..."
```

```
if ~isfolder(main_folder)
    error('please enter a valid directory')
elseif isempty(paths)
    error('please ensure the folder contains .xml ParticleTrack exports')
end
```

If you want to scale the tracks, set the scaling factor here in pixels/distance unit. Otherwise, set scaling to -1 to avoid scaling.

```
scaling = 1 % units per input unit
```

```
scaling = 1
```

Define the units

```
SpaceUnits = 'microns';
```

```
TimeUnits = 'seconds';
```

Enter the timestep in time units. Ensure that you use the inverse of the fps, not exposure.

```
dt=0.3375 % 1/(fps)
```

```
dt = 0.3375
```

what is the minimum track length you want to consider?

```
MinTrackLength = 7
```

```
MinTrackLength = 7
```

Now, generate the MSD analyzer structure. Note that this filters out any tracks with a track length of less than 7 by default, though this can be changed.

```
ma1 = TrackMateImport(main_folder, true, MinTrackLength, scaling);
```

```
Warning: Directory already exists.  
found 588 tracks in the file.  
found 621 tracks in the file.  
found 981 tracks in the file.  
found 511 tracks in the file.  
found a total of 511 tracks in the directory  
Warning: Scaling factor active!  
Warning: plotting only tracks longer than threshold length  
Computing MSD of 1722 tracks... 1Done.
```

```
ma1 = ma1.fitMSD(0.25); % set to short clipping factor to find diffusive rate
```

```
Fitting 1722 curves of MSD = f(t), taking only the first 25% of each curve... 1Done.
```

```
ma1 = ma1.fitLogLogMSD(0.75); % set to long clipping factor for linearity measure
```

```
Fitting 1722 curves of log(MSD) = f(log(t)), taking only the first 75% of each curve... Done.
```

```
d_est = mean(ma1.lfit.a) * (1/dt); % starts with units^2 per frame, divide by  
frames/time unit.  
% This gives the estimated slope of the MSD in units^2/second  
gamma_est = mean(ma1.loglogfit.alpha); % Gives time scaling factor of the MSD.  
  
fprintf(strcat("The diffusion coefficient is D=", string(d_est), ' ', SpaceUnits,  
'^2/', TimeUnits));
```

```
The diffusion coefficient is D=1.5767microns^2/seconds
```

```
fprintf(strcat("The time scaling factor is gamma=", string(gamma_est), '.'));
```

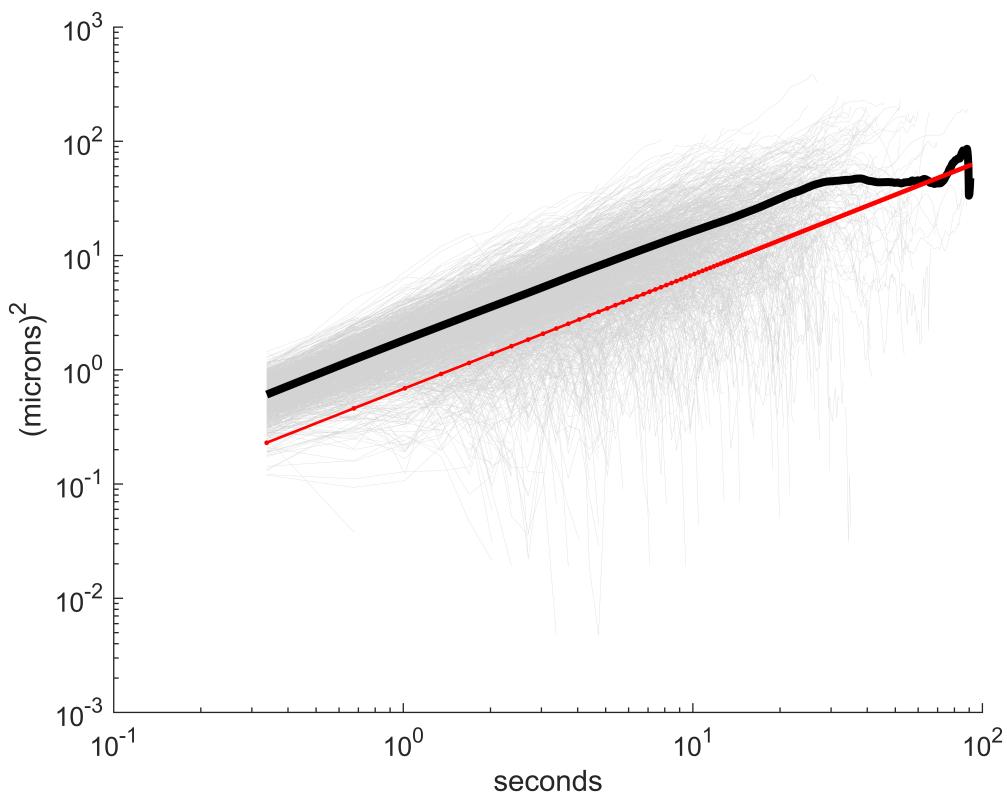
```
The time scaling factor is gamma=0.82001.
```

Now, we plot the MSD. If you want to plot an expected value, set it. Otherwise, set D to -1.

```
D_expect=0.17021;  
fprintf(strcat('expected diffusion coefficient: ', string(D_expect), ' (',  
SpaceUnits, ')^2/', TimeUnits))
```

```
expected diffusion coefficient:0.17021 (microns)^2/seconds
```

```
FigMeanMSD(SpaceUnits, TimeUnits,ma1, dt,D_expect,true)
```

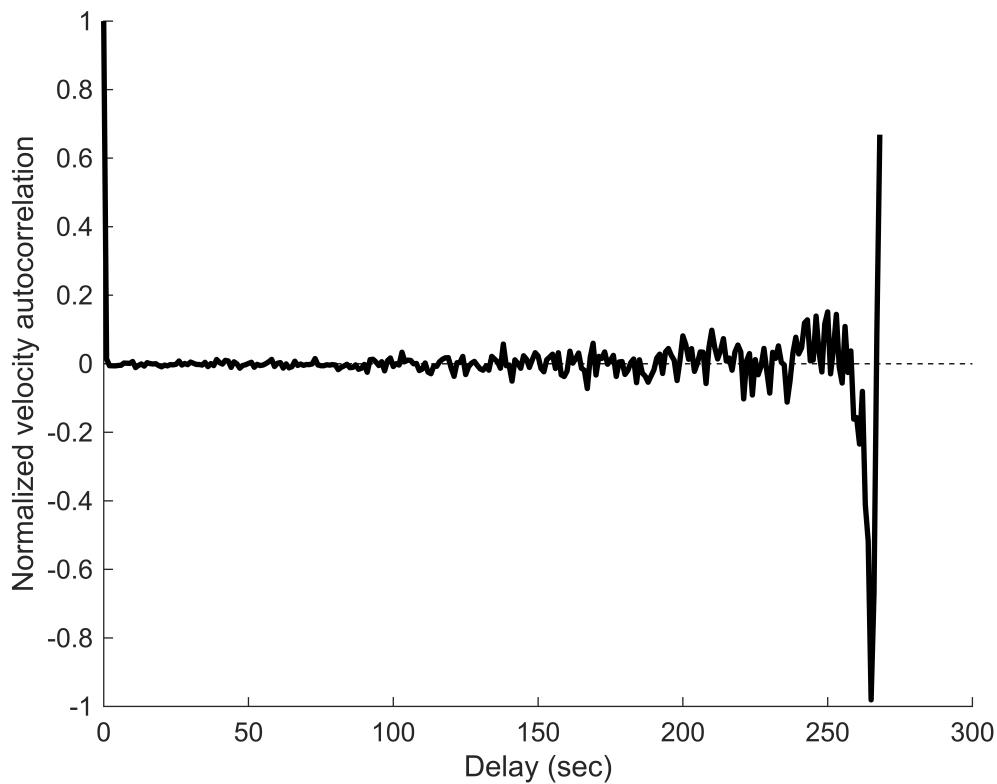


```
ans = 270x5
     0         0         0    911.2509         0
0.3375  0.6065  0.1222   872.2805  0.2298
0.6750  1.2255  0.2848   835.5401  0.4596
1.0125  1.8304  0.4830   806.0014  0.6894
1.3500  2.4199  0.7115   779.6502  0.9191
1.6875  3.0040  0.9611   753.4690  1.1489
2.0250  3.5799  1.2276   727.7683  1.3787
2.3625  4.1456  1.4923   702.2901  1.6085
2.7000  4.7146  1.7700   677.3533  1.8383
3.0375  5.2876  2.0678   654.1510  2.0681
.
.
```

Now we create a plot of the mean velocity correlation as a function of time.

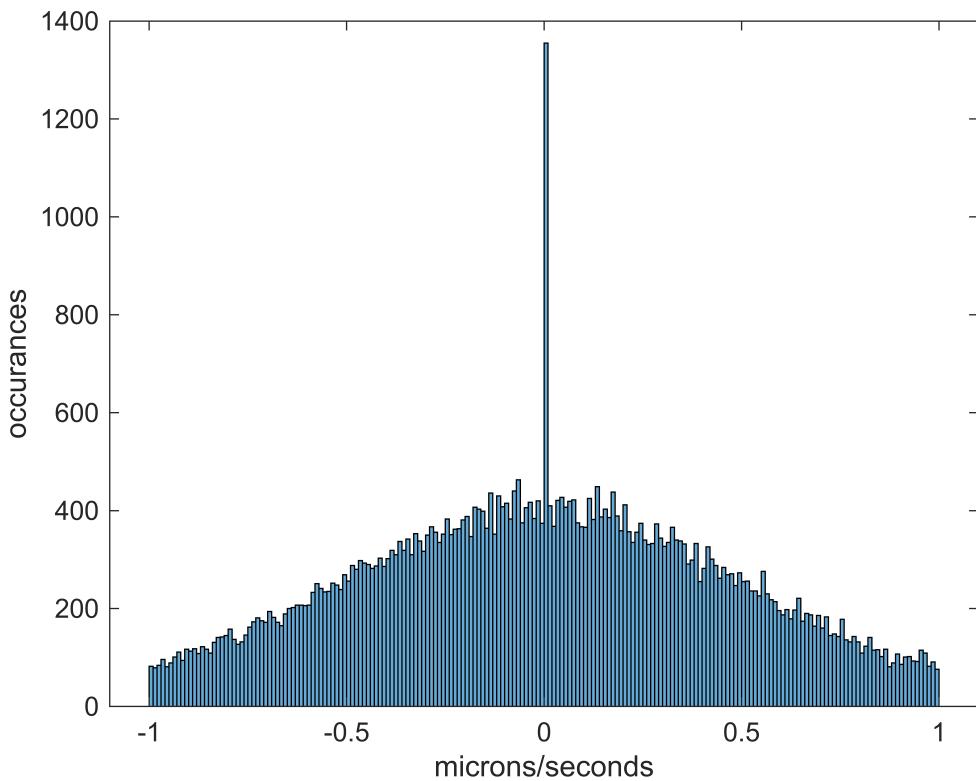
```
figure;
ma1.plotMeanVCorr;
```

```
Computing velocity autocorrelation of 1722 tracks... Done.
```



We may also directly compute velocities and plot them as a histogram

```
v = ma1.getVelocities;
V=vertcat(v{:});
edges2 = -1:0.01:1;
histogram(V(:,2),edges2)
xlabel(strcat(SpaceUnits, '/', TimeUnits))
ylabel("occurrences")
```



Now, we must call CenterTracks.

```
CenterTracks=CreateCenterTracks(ma1.tracks);
```

We may now create a short vs long track diagram.

```
figure();
% UniDomainFigCenterJuxt(SpaceUnits, CenterTracks, 10,10, 10, 10, true, true, 1)
```

Now create a VanHovePlot. This function calls the new VanHove2.m so it is relatively fast and creates bins with equal widths, but it can still take a while to run.

Now, we can can manimputme the CreateVanHovePlots function in a few ways.

We can change the time steps, the number of particles assigned to each bin, and the minimum bin width.

The BinSize determines the number of particles which the Van

```
BinSize = 10
```

```
BinSize = 10
```

```
MinBin = 0.01 % In microns
```

```
MinBin = 0.0100
```

```
VanHoveStats= CreateVanHovePlots(SpaceUnits, TimeUnits, ma1.tracks, BinSize,
[1,2,3,5], main_folder, MinBin, dt)
```

Calculating VanHove Distribution dt=1

Calculating VanHove Distribution dt=2

Calculating VanHove Distribution dt=3

Calculating VanHove Distribution dt=5

Cleaning up for dt=1

Cleaning up for dt=2

Cleaning up for dt=3

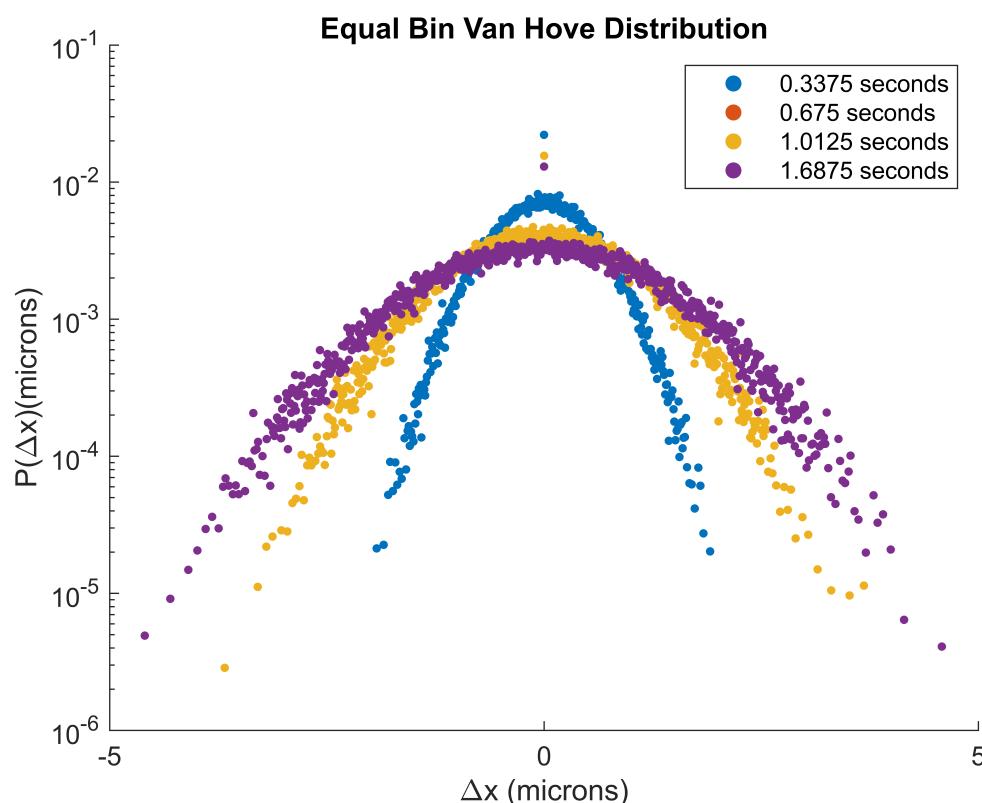
Cleaning up for dt=5

Sorting dt=1 into bins

Sorting dt=2 into bins

Sorting dt=3 into bins

Sorting dt=5 into bins



Warning: Over 250 bins. Irregularities may occur.

VanHoveStats = struct with fields:

 Data: {5×1 cell}

 CenterPoint: {5×1 cell}

 EquiProb: {5×1 cell}

 TotStepsCount: [4×2 table]

 BinSize: 10

 tau: [1 2 3 5]

 FrameTime: 0.3375

 MinBin: 0.0100

Save the data in SuperStruct format to the SuperStructs Github folder.

```
MasterSaveD(monkier, ma1,main_folder, MinTrackLength, SpaceUnits, TimeUnits,  
VanHoveStats, cd)
```

```
ans =  
"C:\Users\al3xm\Documents\GitHub\HISTia\SuperStructs\1um beads in water at 3.33fps2025-03-20.21.05.43.mat"
```

Save this output as a pdf for later viewing.