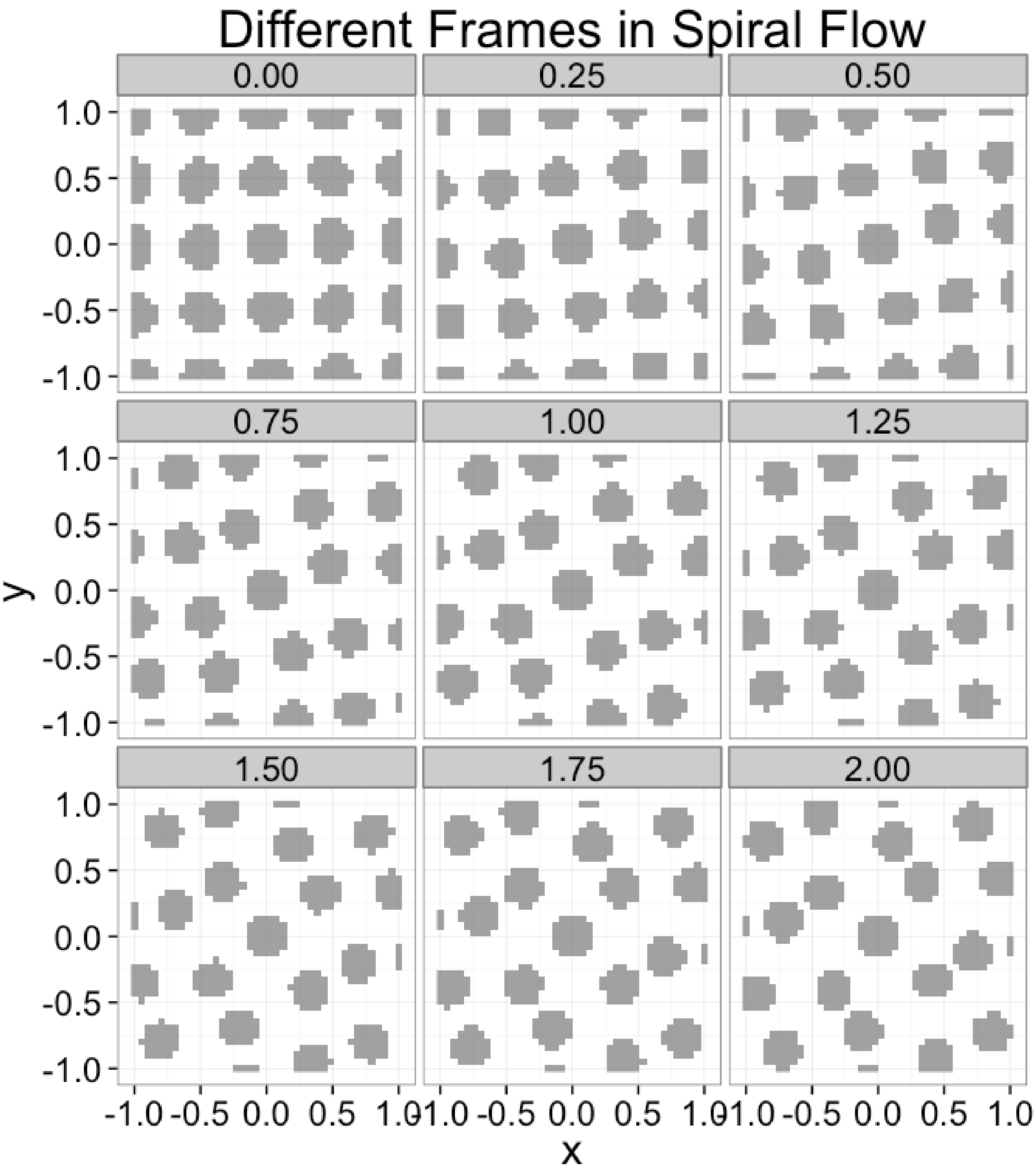

Different Frames in Spiral Flow
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
1.0
0.5
0.0
-0.5
-1.0
y
-1.0 -0.5 0.0 0.5 1.0
x

Quantitative Big Imaging

author: Kevin Mader date: 17 April 2014 width: 1440 height: 900 css: ../template.css transition: rotate

ETHZ: 227-0966-00L

Dynamic Experiments

Course Outline

- 20th February - Introductory Lecture
- 27th February - Filtering and Image Enhancement (A. Kaestner)
- 6th March - Basic Segmentation, Discrete Binary Structures
- 13th March - Advanced Segmentation
- 20th March - Analyzing Single Objects
- 27th March - Analyzing Complex Objects
- 3rd April - Many Objects and Distributions
- 10th April - Statistics and Reproducibility
- 17th April - **Dynamic Experiments**
- 8th May - Scaling Up / Big Data
- 15th May - Guest Lecture - In-Operando Imaging of Batteries (V. Wood)
- 22th May - Project Presentations

Literature / Useful References

Books

- Jean Claude, Morphometry with R
  - Online (http://link.springer.com/book/10.1007%2F978-0-387-77789-4) through ETHZ
  - Buy it (http://www.amazon.com/Morphometrics-R-Use-Julien-Claude/dp/038777789X)
- John C. Russ, “The Image Processing Handbook”,(Boca Raton, CRC Press)
  - Available online (http://dx.doi.org/10.1201/9780203881095) within domain ethz.ch (or proxy.ethz.ch / public VPN)

Papers / Sites

- Comparsion of Tracking Methods in Biology
    - Chenouard, N., Smal, I., de Chaumont, F., Maška, M., Sbalzarini, I. F., Gong, Y., … Meijering, E. (2014). Objective comparison of particle tracking methods. Nature Methods, 11(3), 281–289. doi:10.1038/nmeth.2808
    - Maska, M., Ulman, V., Svoboda, D., Matula, P., Matula, P., Ederra, C., … Ortiz-de-Solorzano, C. (2014). A benchmark for comparison of cell tracking algorithms. Bioinformatics (Oxford, England), btu080–. doi:10.1093/bioinformatics/btu080
- Multiple Hypothesis Testing
    - Coraluppi, S. & Carthel, C. Multi-stage multiple-hypothesis tracking. J. Adv. Inf. Fusion 6, 57–67 (2011).
    - Chenouard, N., Bloch, I. & Olivo-Marin, J.-C. Multiple hypothesis tracking in microscopy images. in Proc. IEEE Int. Symp. Biomed. Imaging 1346–1349 (IEEE, 2009).

Previously on QBI ...

- Image Enhancment
    - Highlighting the contrast of interest in images
    - Minimizing Noise
- Understanding image histograms
- Automatic Methods
- Component Labeling
- Single Shape Analysis
- Complicated Shapes
- Distribution Analysis

Quantitative "Big" Imaging

The course has covered imaging enough and there have been a few quantitative metrics, but "big" has not really entered.

What does **big** mean?

- Not just / even large
- it means being ready for *big data*
- volume, velocity, variety (3 V's)
- scalable, fast, easy to customize

So what is "big" imaging

- doing analyses in a disciplined manner

  - fixed steps
  - easy to regenerate results
  - no *magic*
- having everything automated
  - 100 samples is as easy as 1 sample
- being able to adapt and reuse analyses
  - one really well working script and modify parameters
  - different types of cells
  - different regions

Objectives

1. What sort of dynamic experiments do we have?
2. How can we design good dynamic experiments?
3. How can we track objects between points?
   - How can we track shape?
   - How can we track distribution?
4. How can we track topology?
5. How can we track voxels?
6. How can we assess deformation and strain?
7. How can assess more general cases?

Outline

- Motivation (Why and How?)
- Scientific Goals
- Experiments
  - Simulations
  - Experiment Design
- Object Tracking
- Distribution
- Topology

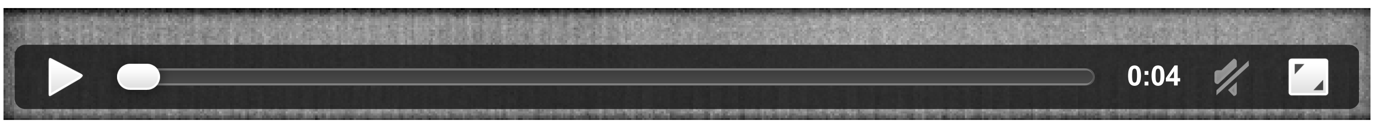
- Voxel-based Methods
  - Cross Correlation
  - DIC
  - DIC + Physics
- General Problems
  - Thickness - Lung Tissue
  - Curvature - Metal Systems

- Two Point Correlation - Volcanic Rock

Motivation

- 3D images are already difficult to interpret on their own
- 3D movies (4D) are almost impossible

- 2D movies (3D) can also be challenging



We can say that it looks like, but many pieces of quantitative information are difficult to extract

- how fast is it going?
- how many particles are present?
- are their sizes constant?
- are some moving faster?
- are they rearranging?

Scientific Goals

Rheology

Understanding the flow of liquids and mixtures is important for many processes

- blood movement in arteries, veins, and capillaries
- oil movement through porous rock
- air through dough when cooking bread
- magma and gas in a volcano

Deformation

Deformation is similarly important since it plays a significant role in the following scenarios

- red blood cell lysis in artificial heart valves
- microfractures growing into stress fractures in bone
- toughening in certain wood types

Experiments

The first step of any of these analyses is proper experimental design. Since there is always

- a limited field of view
- a voxel size
- a maximum rate of measurements
- a non-zero cost for each measurement

There are always trade-offs to be made between getting the best possible high-resolution nanoscale dynamics and capturing the system level behavior.

- If we measure too fast
    - sample damage
    - miss out on long term changes
    - have noisy data
- Too slow
    - miss small, rapid changes
    - blurring and other motion artifacts
- Too high resolution
    - not enough unique structures in field of view to track
- Too low resolution
    - not sensitive to small changes

Simulation

In many cases, experimental data is inherited and little can be done about the design, but when there is still the opportunity, simulations provide a powerful tool for tuning and balancing a large number parameters

Simulations also provide the ability to pair post-processing to the experiments and determine the limits of tracking.

What do we start with?

Going back to our original cell image

1. We have been able to get rid of the noise in the image and find all the cells (lecture 2-4)
2. We have analyzed the shape of the cells using the shape tensor (lecture 5)
3. We even separated cells joined together using Watershed (lecture 6)
4. We have created even more metrics characterizing the distribution (lecture 7)

We have at least a few samples (or different regions), large number of metrics and an almost as large number of parameters to *tune*

How do we do something meaningful with it?

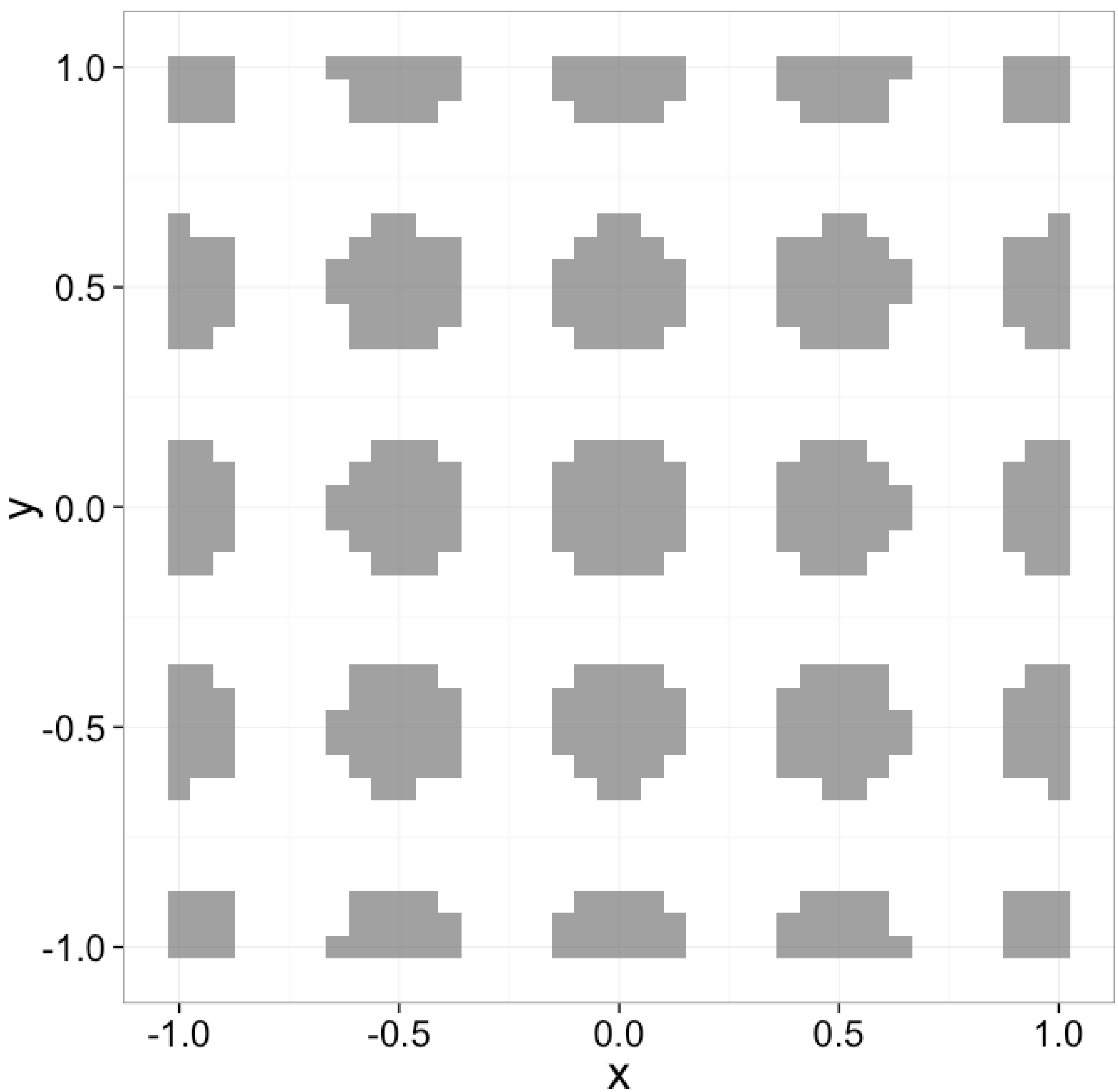
Basic Simulation

We start with a starting image

+/- R Code

- A number of sphere objects with the same radius scattered evenly across the field of view

  +/- R Code

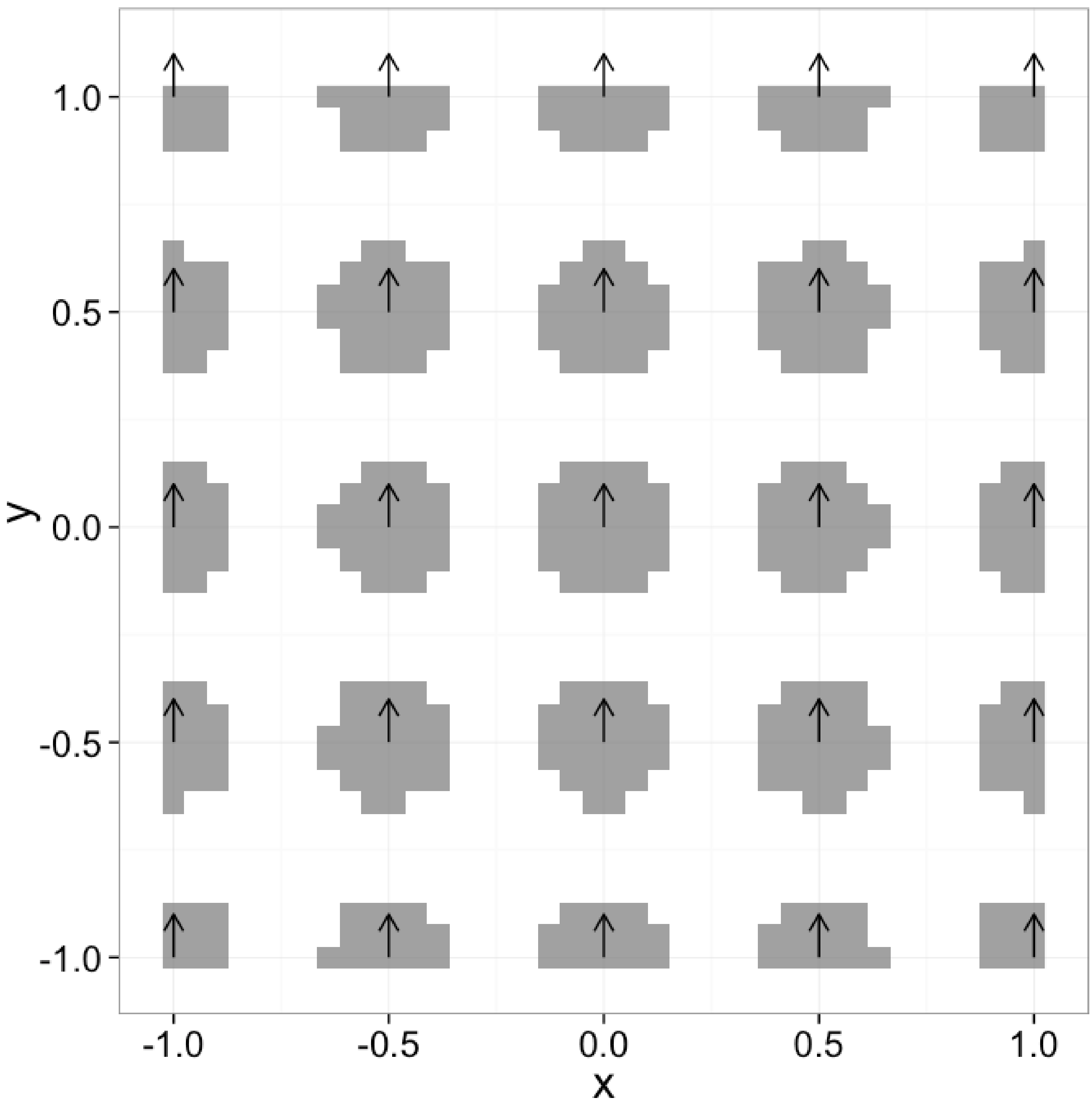


Analysis

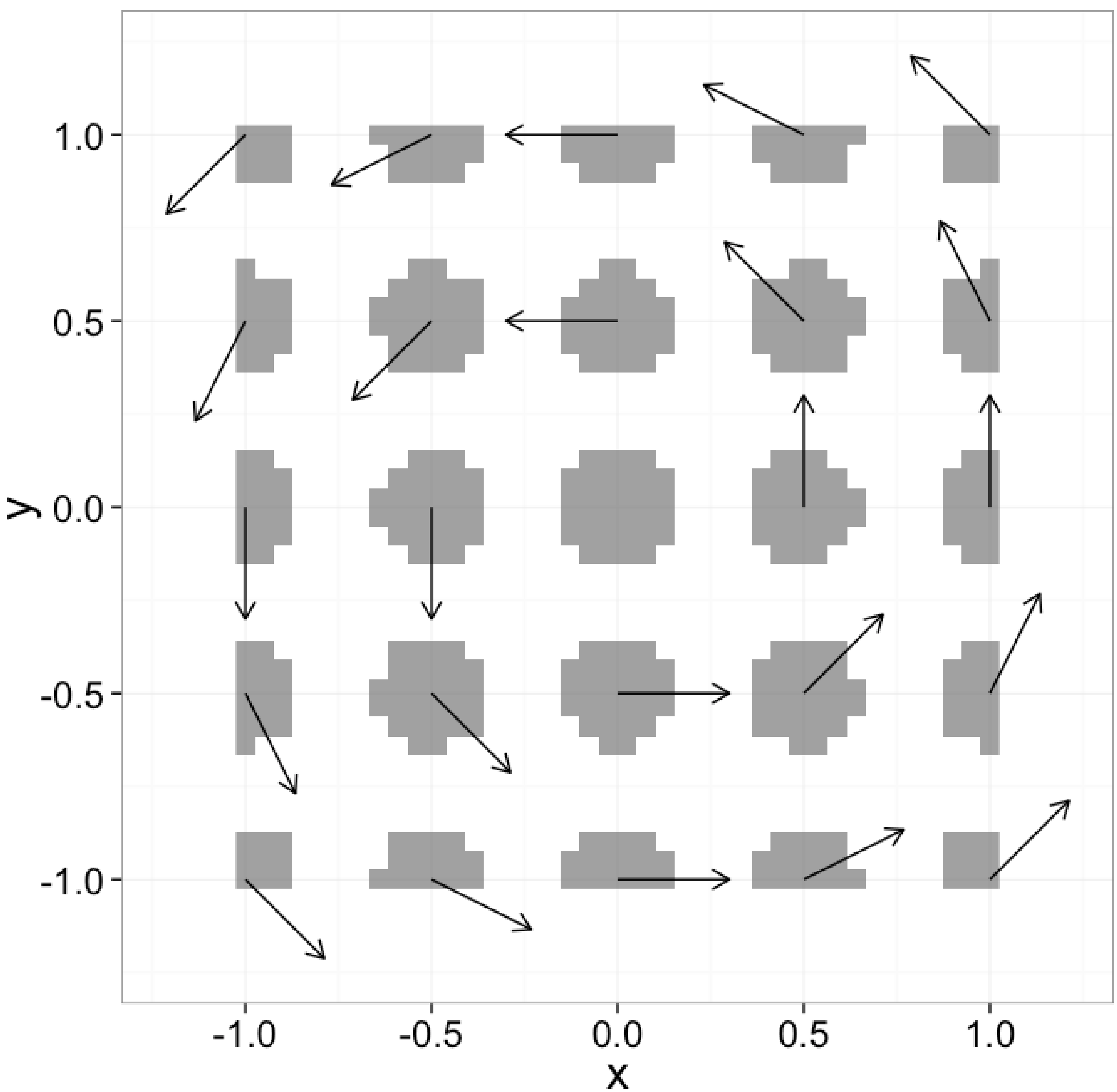
- Threshold
- Component Label
- Shape Analysis
- Distribution Analysis

Describing Motion

$$\vec{v}(\vec{x}) = \langle 0, 0.1 \rangle$$

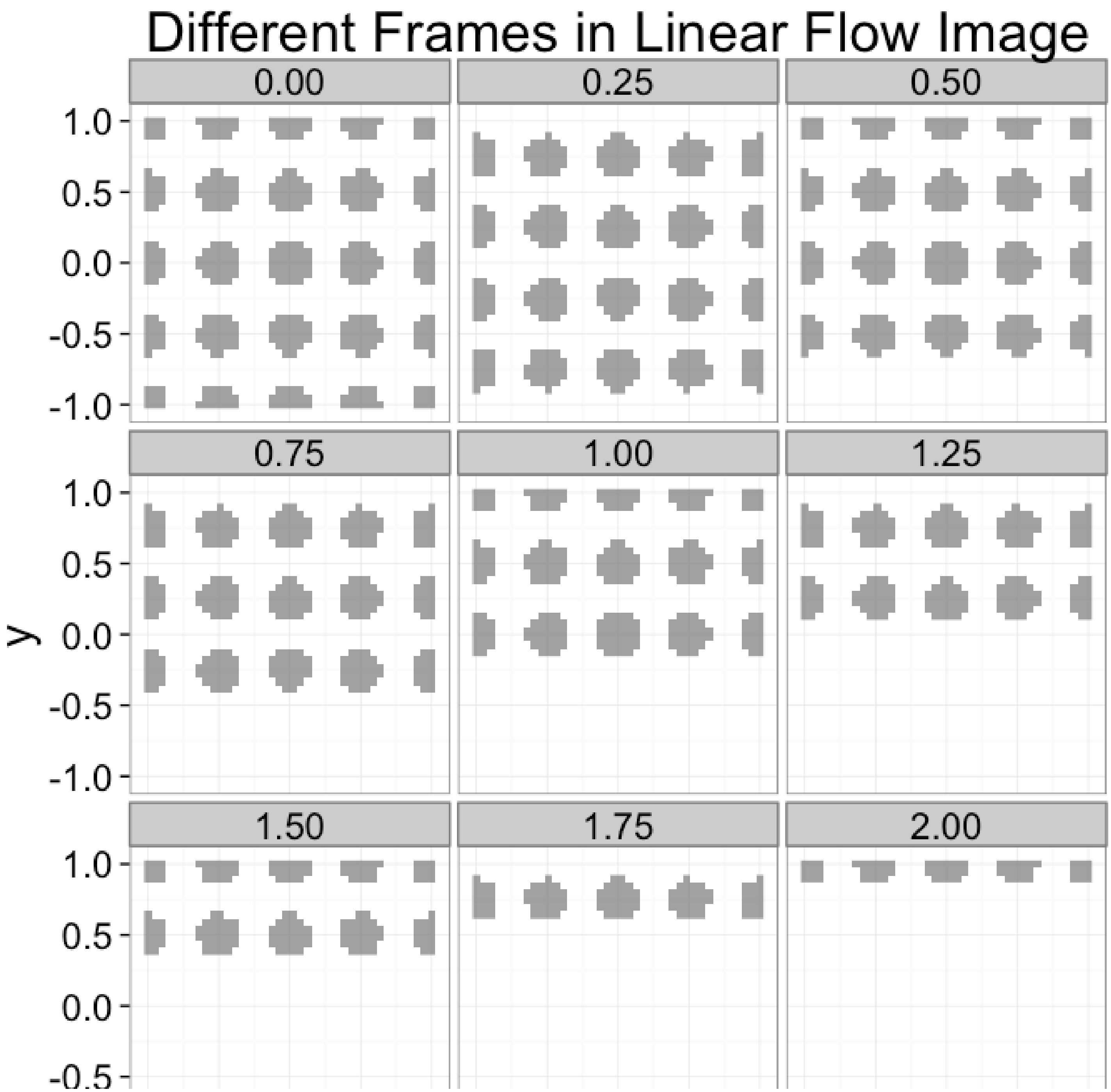


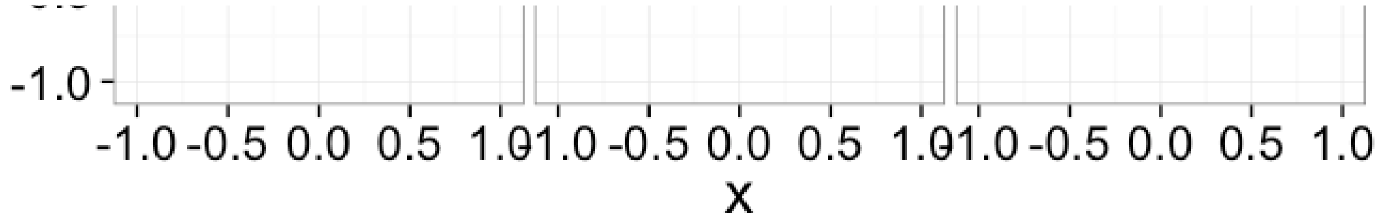
$$\vec{v}(\vec{x}) = 0.3\,\frac{\vec{x}}{\|\vec{x}\|} \times \langle 0, 0, 1 \rangle$$



Many Frames

$$\vec{v}(\vec{x}) = \langle 0, 0.1 \rangle$$





$$\vec{v}(\vec{x}) = 0.3\,\frac{\vec{x}}{||\vec{x}||} \times \langle 0, 0, 1 \rangle$$

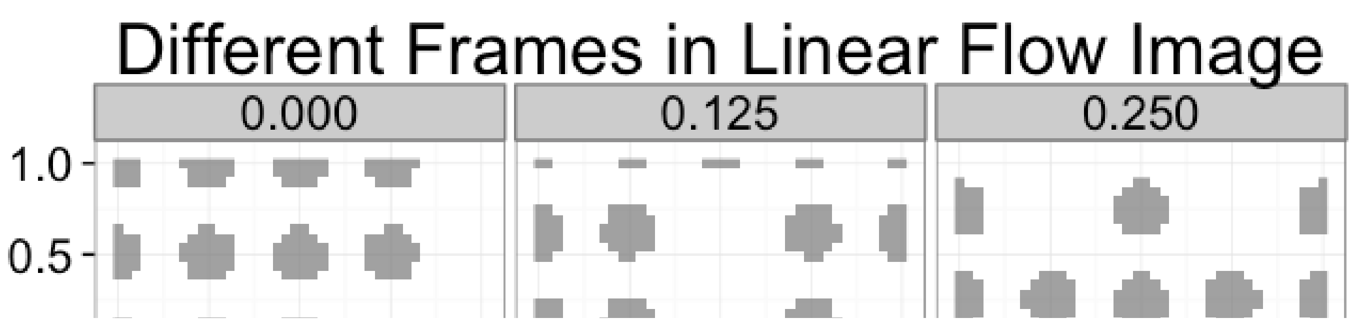
+/- R Code

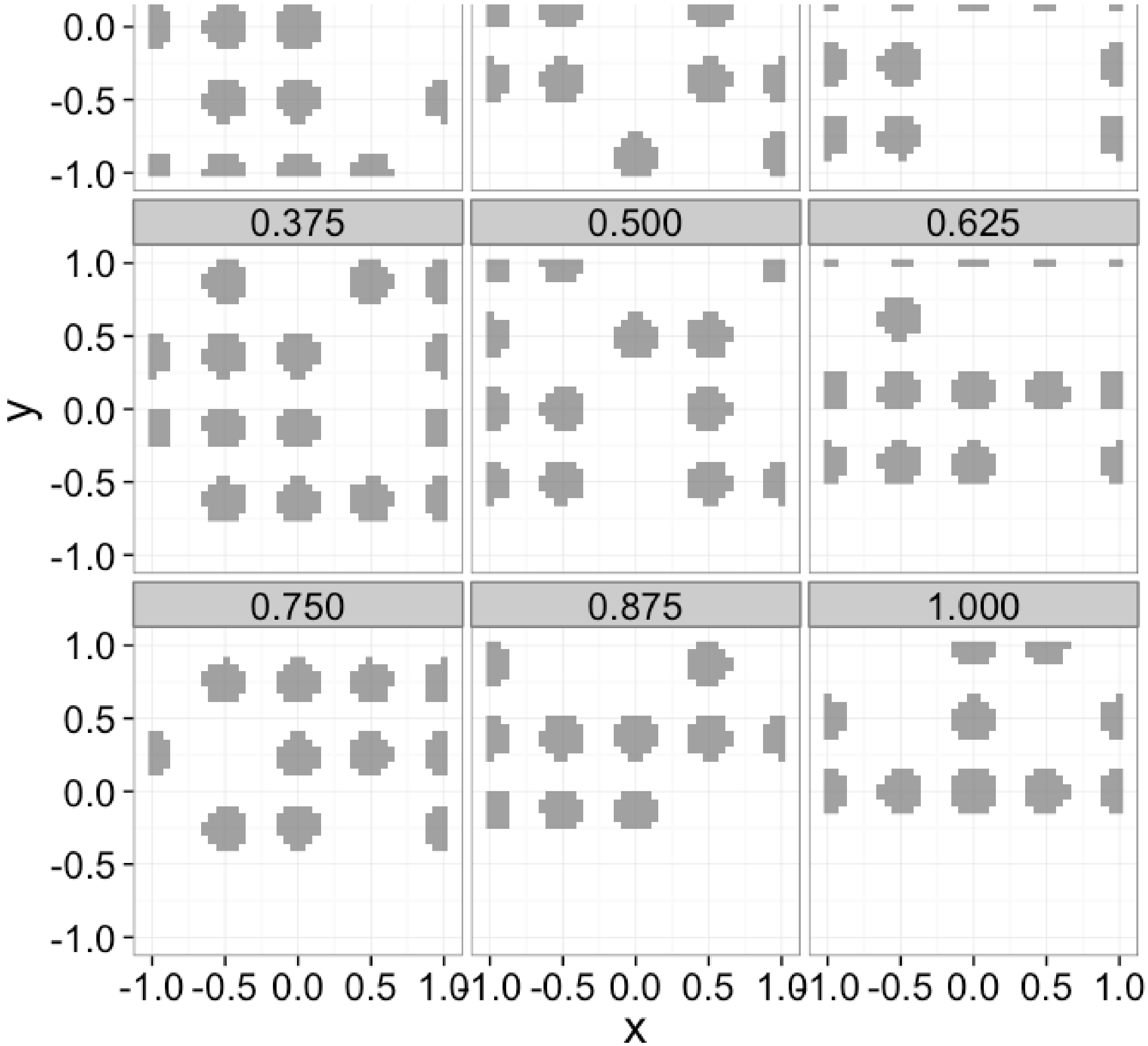
Random Appearance / Disappearance

Under perfect imaging and experimental conditions objects should not appear and reappear but due to

- noise
- limited fields of view / depth of field
- discrete segmentation approachs
- motion artifacts
    - blurred objects often have lower intensity values than still objects

It is common for objects to appear and vanish regularly in an experiment.

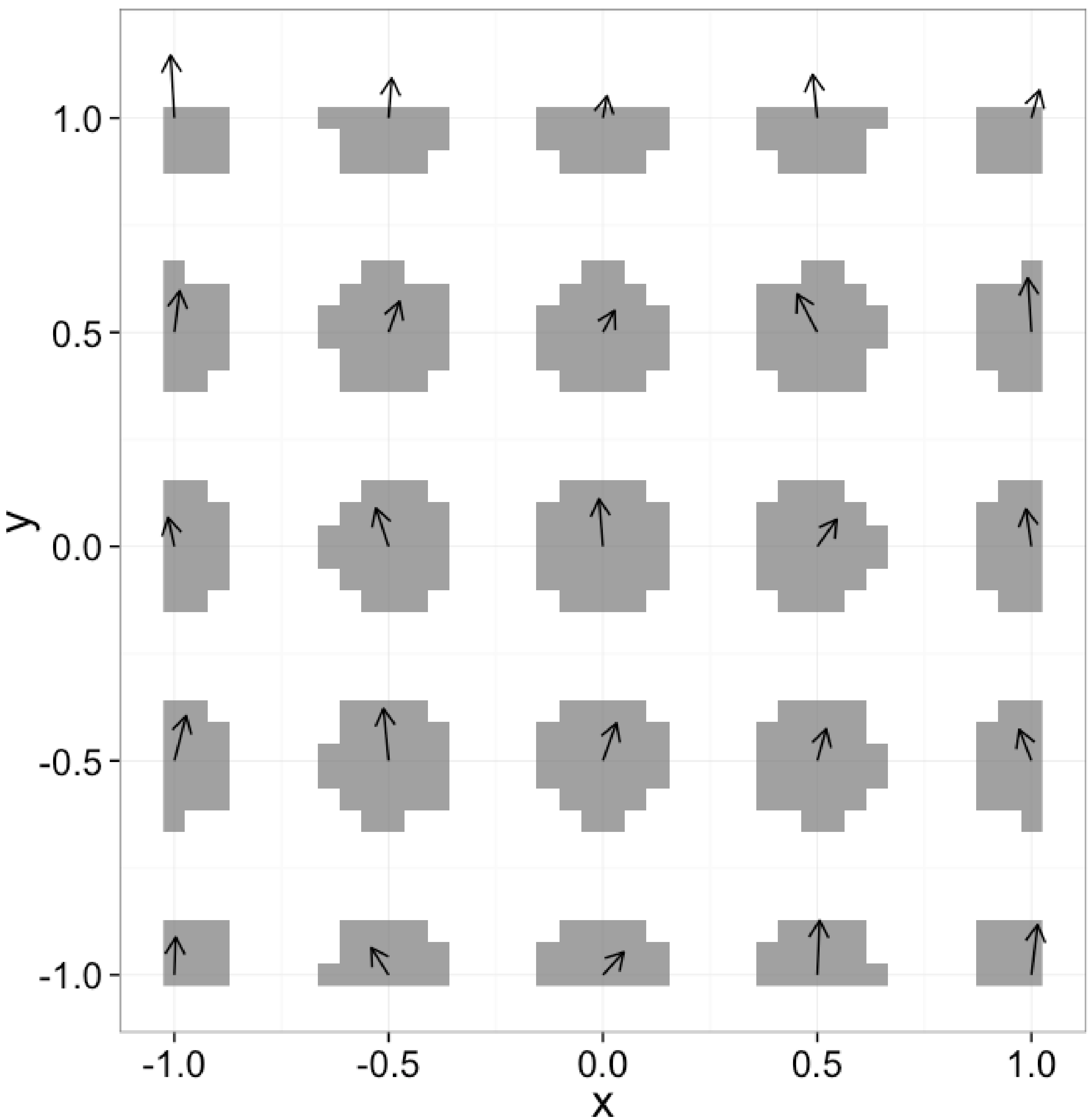




Jitter / Motion Noise

Even perfect spherical objects do not move in a straight line. The jitter can be seen as a stochastic variable with a random magnitude (a) and angle (b). This is then sampled at every point in the field

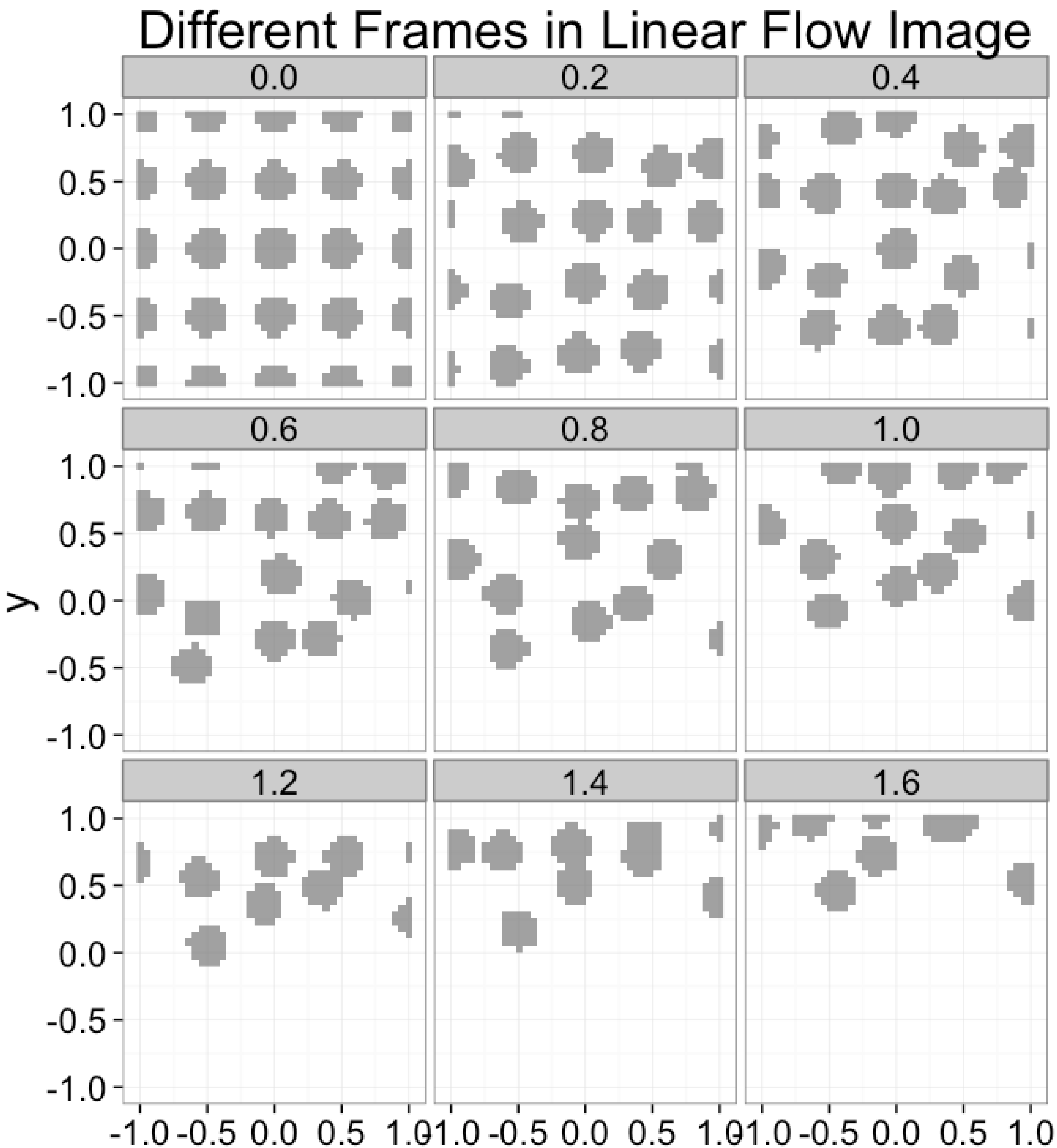
$$\vec{v}(\vec{x}) = \vec{v}_L(\vec{x}) + ||a||\measuredangle b$$



+/- R Code

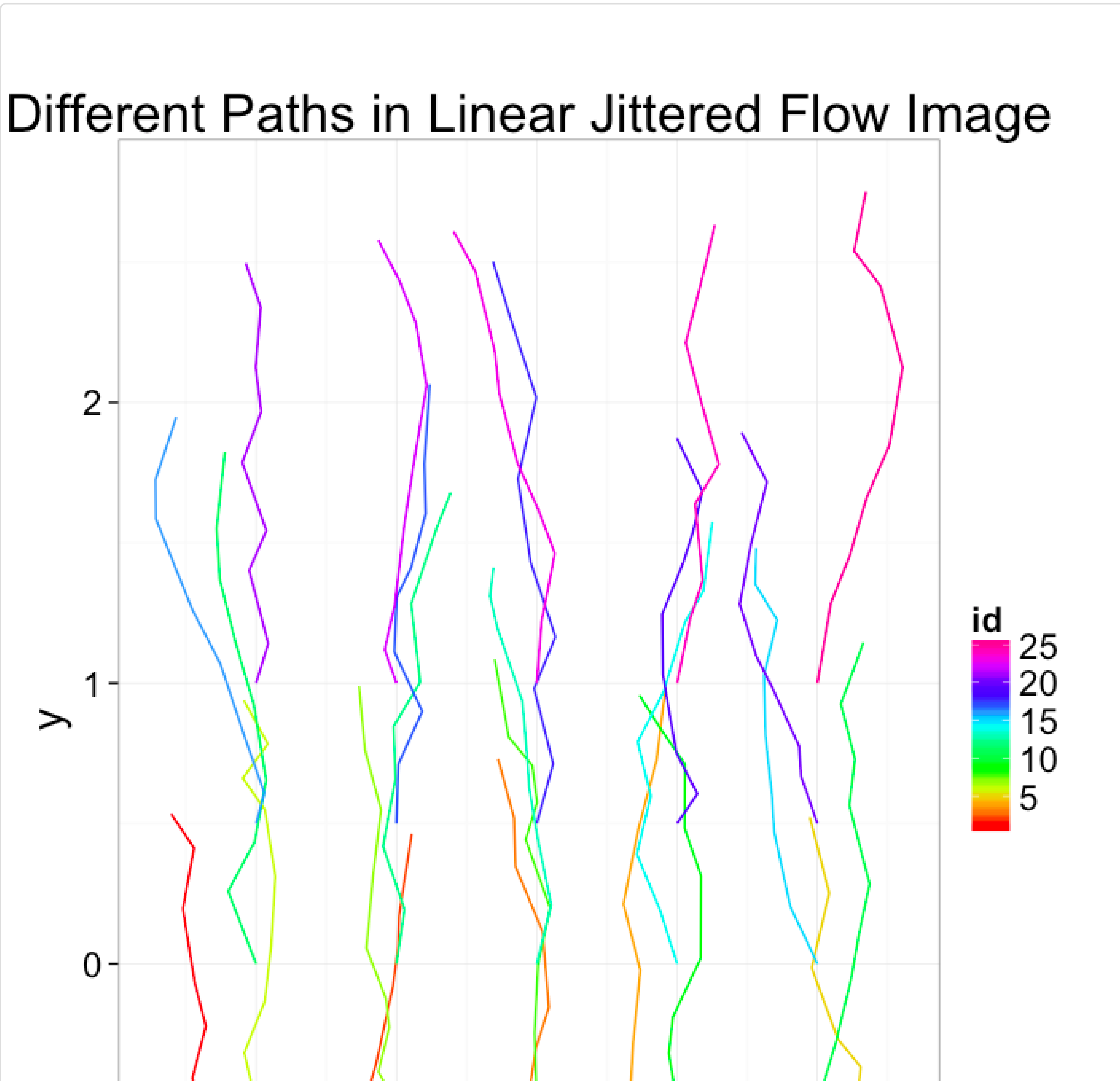
Jitter (Continued)

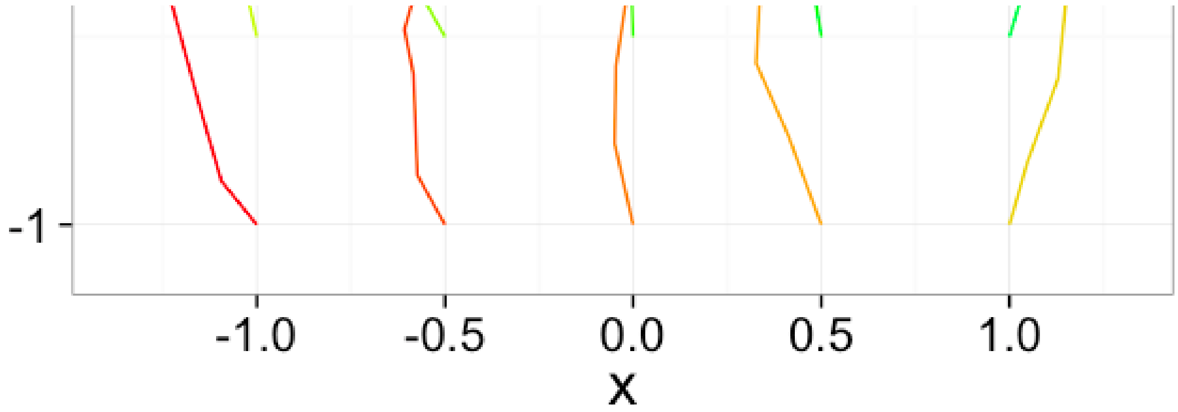
Over many frames this can change the path significantly



x

The simulation can be represented in a more clear fashion by using single lines to represent each spheroid





Limits of Tracking

We see that visually tracking samples can be difficult and there are a number of parameters which affect the ability for us to clearly see the tracking.

- flow rate
- flow type
- density
- appearance and disappearance rate
- jitter
- particle uniqueness

We thus try to quantify the limits of these parameters for different tracking methods in order to design experiments better.

Acquisition-based Parameters

- acquisition rate → flow rate, jitter (per frame)
- resolution → density, appearance rate

Experimental Parameters

- experimental setup (pressure, etc) → flow rate/type
- polydispersity → particle uniqueness
- vibration/temperature → jitter
- mixture → density

Tracking: Nearest Neighbor

While there exist a number of different methods and complicated approaches for tracking, for experimental design it is best to start with the simplist, easiest understood method. The limits of this can be found and components added as needed until it is possible to realize the experiment

- If a dataset can only be analyzed with a multiple-hypothesis testing neural network model then it might not be so reliable

We then return to *nearest neighbor* which means we track a point ($\vec{P}_0$) from an image (I_0) at t_0 to a point ($\vec{P}_1$) in image (I_1) at t_1 by

$$\vec{P}_1 = \text{argmin}(||\vec{P}_0 - \vec{y}|| \forall \vec{y} \in I_1)$$

Scoring Tracking

In the following examples we will use simple metrics for scoring fits where the objects are matched and the number of misses is counted.

There are a number of more sensitive scoring metrics which can be used, by finding the best submatch for a given particle since the number of matches and particles does not always correspond. See the papers at the beginning for more information

+/- R Code

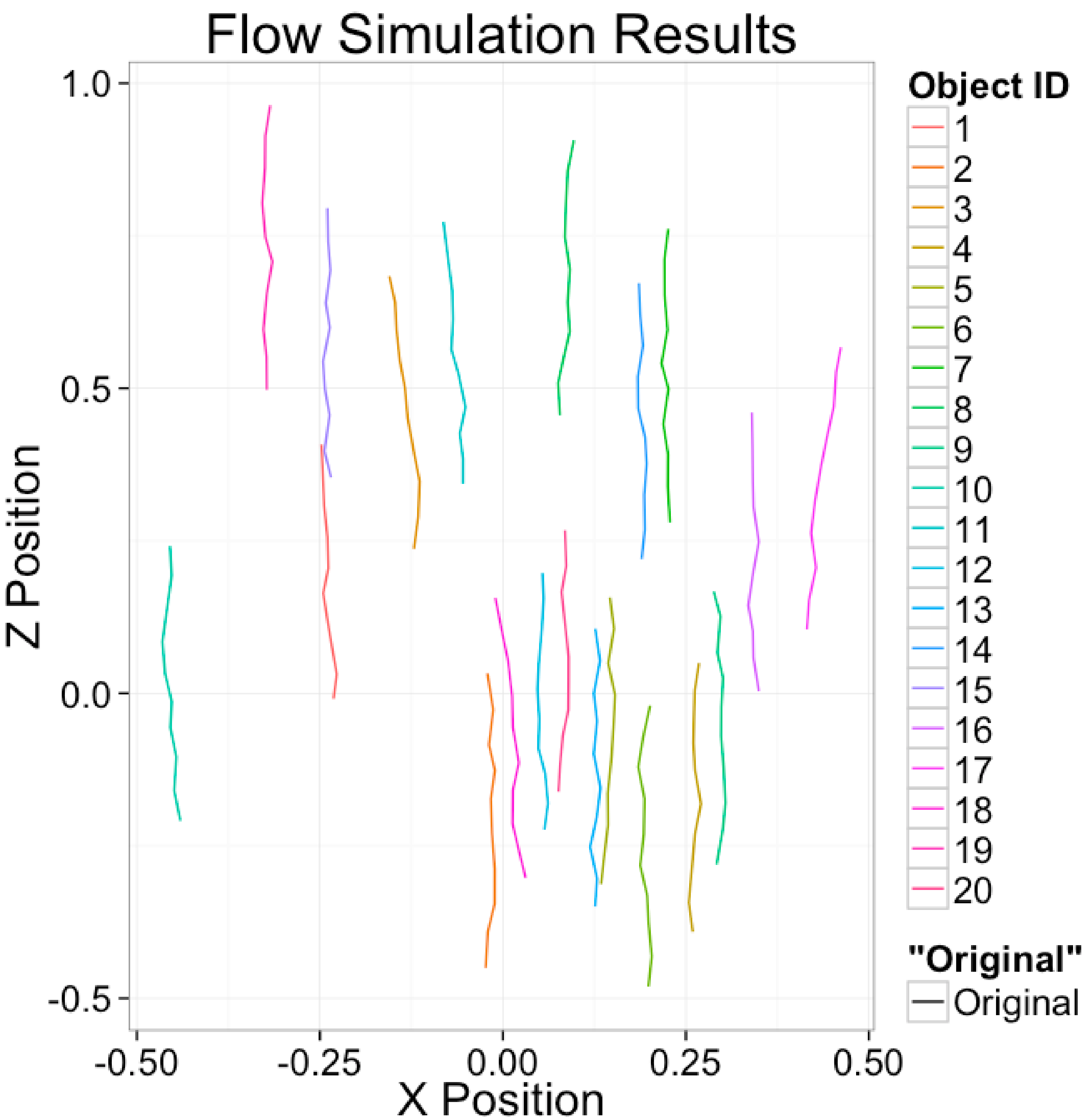
Basic Simulations

Input flow from simulation

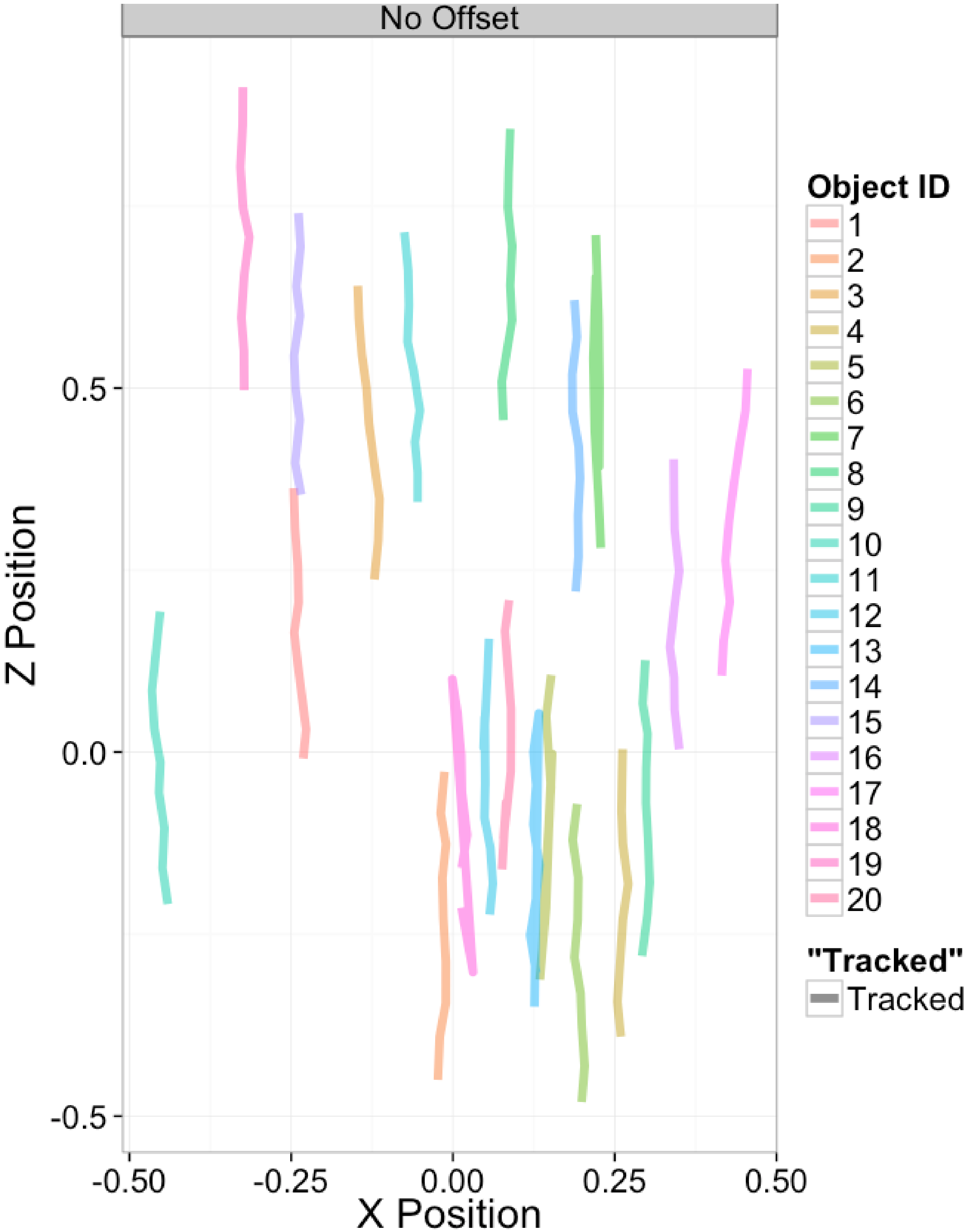
$$\vec{v}(\vec{x}) = \langle 0, 0, 0.05 \rangle + ||0.01|| \measuredangle b$$

+/- R Code

+/- R Code



Nearest Neighbor Tracking


No Offset
Z Position
X Position
0.5
0.0
-0.5
-0.50
-0.25
0.00
0.25
0.50
Object ID
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
"Tracked"
Tracked

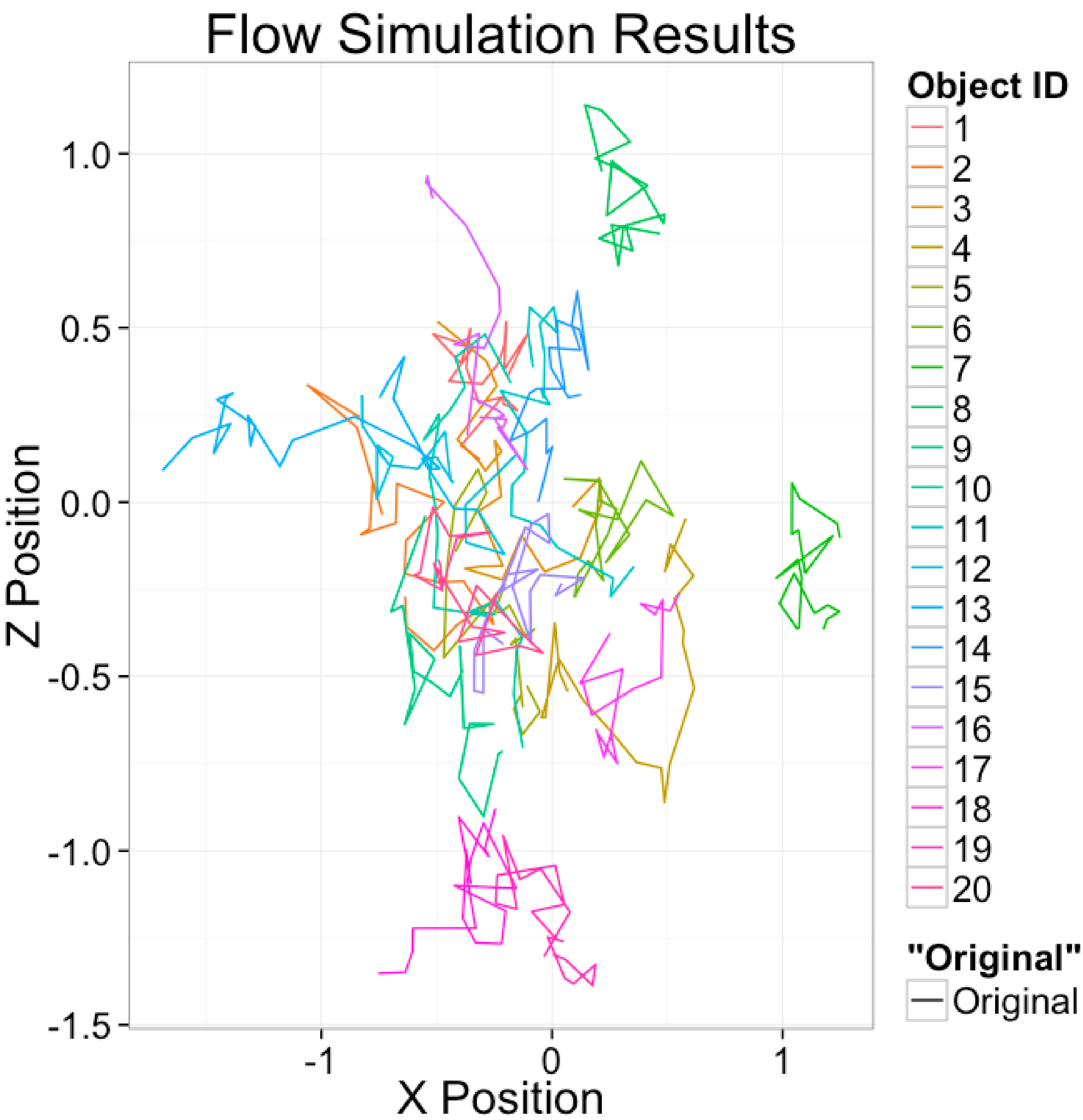
More Complicated Flows

Input flow from simulation

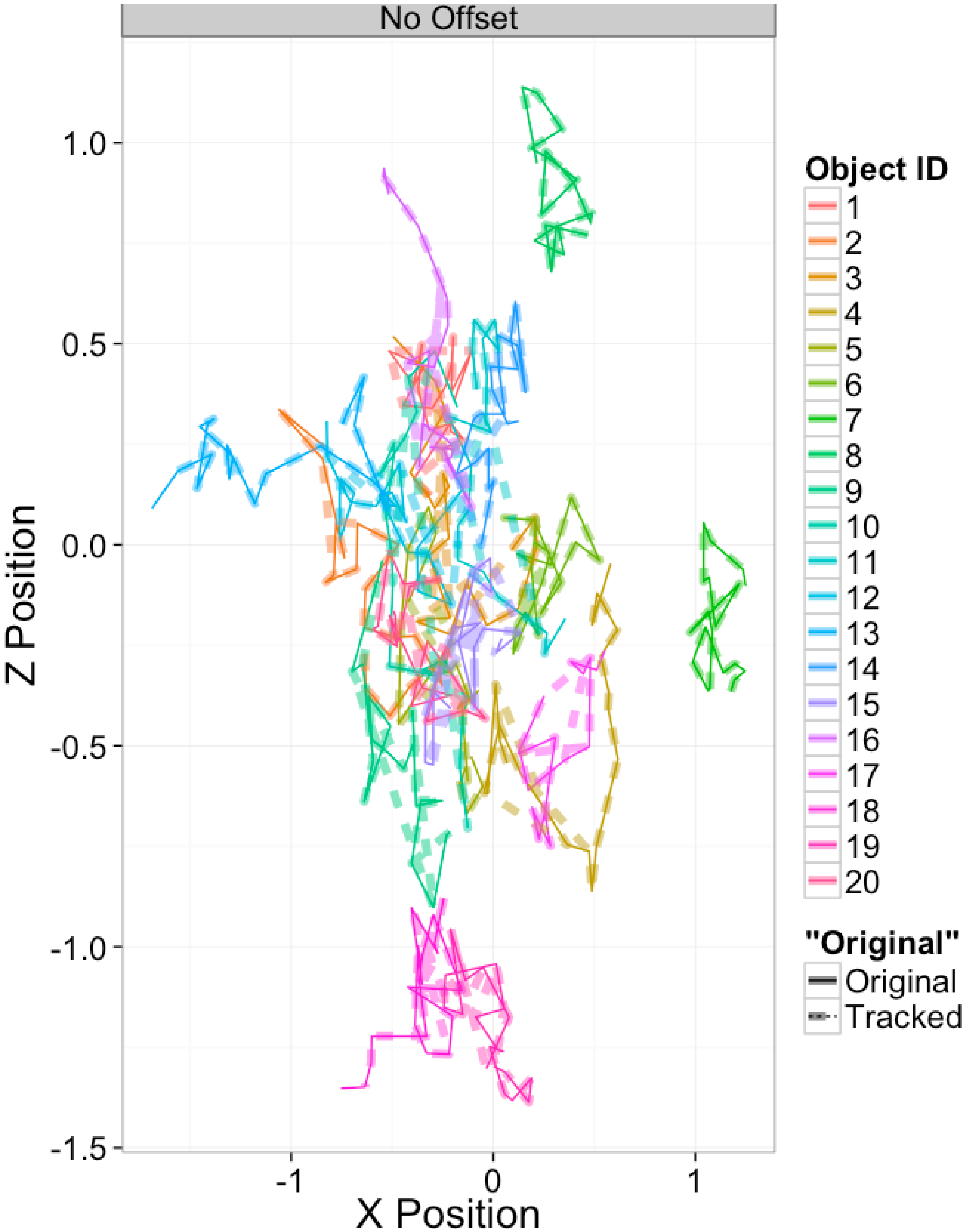
$$\vec{v}(\vec{x}) = \langle 0, 0, 0.01 \rangle + \|0.05\| \measuredangle b$$

+/- R Code

+/- R Code

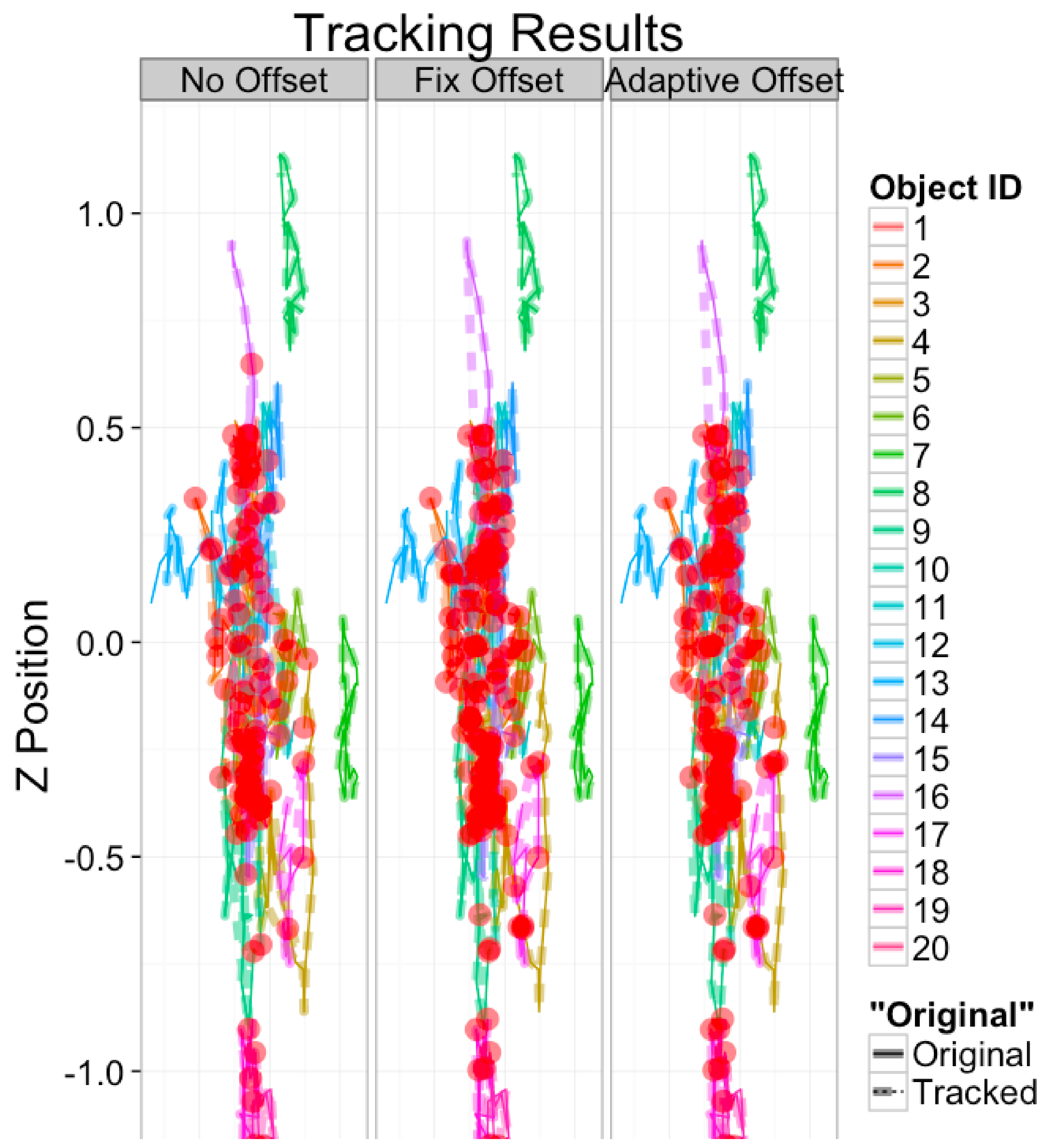


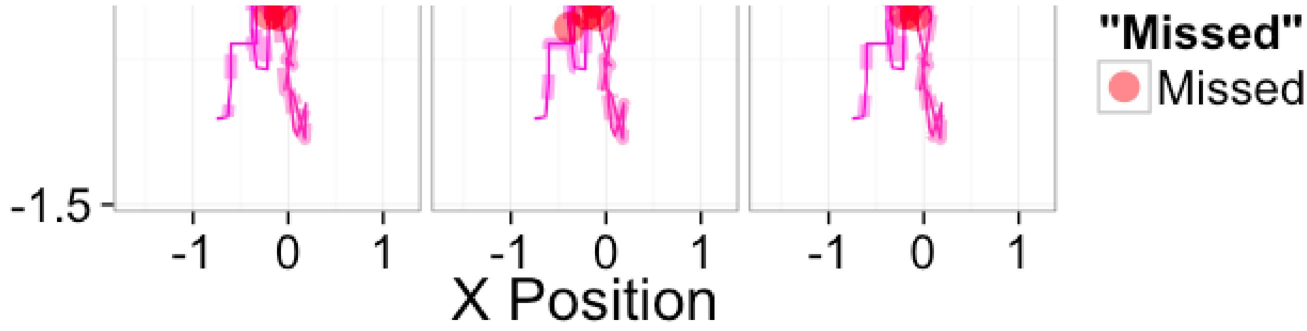
Nearest Neighbor Tracking


No Offset
Z Position
X Position
1.0
0.5
0.0
-0.5
-1.0
-1.5
-1
0
1
Object ID
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
"Original"
Original
Tracked

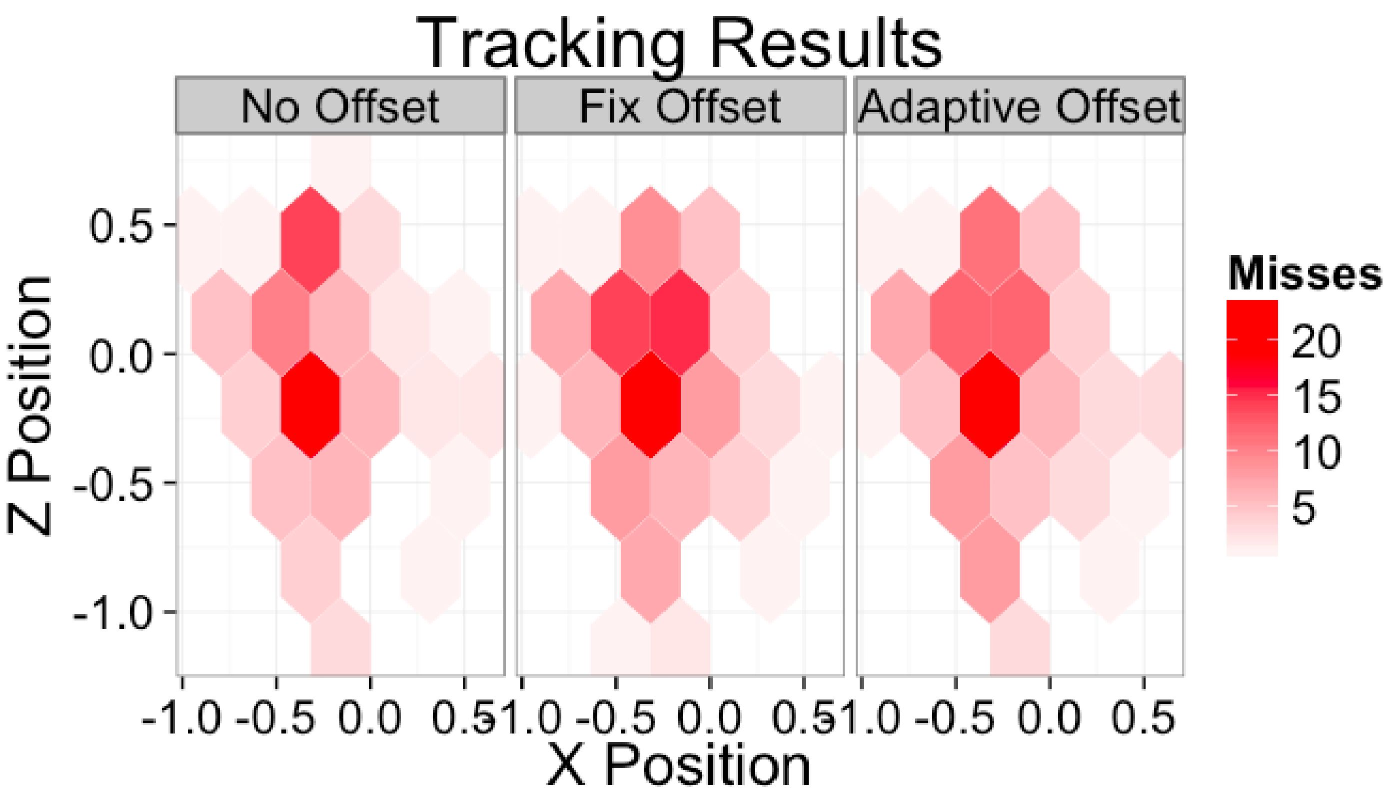
Quantifying Tracking Rate

We can then quantify the success rate of each algorithm on the data set using the very simple match and mismatch metrics





Counting Misses

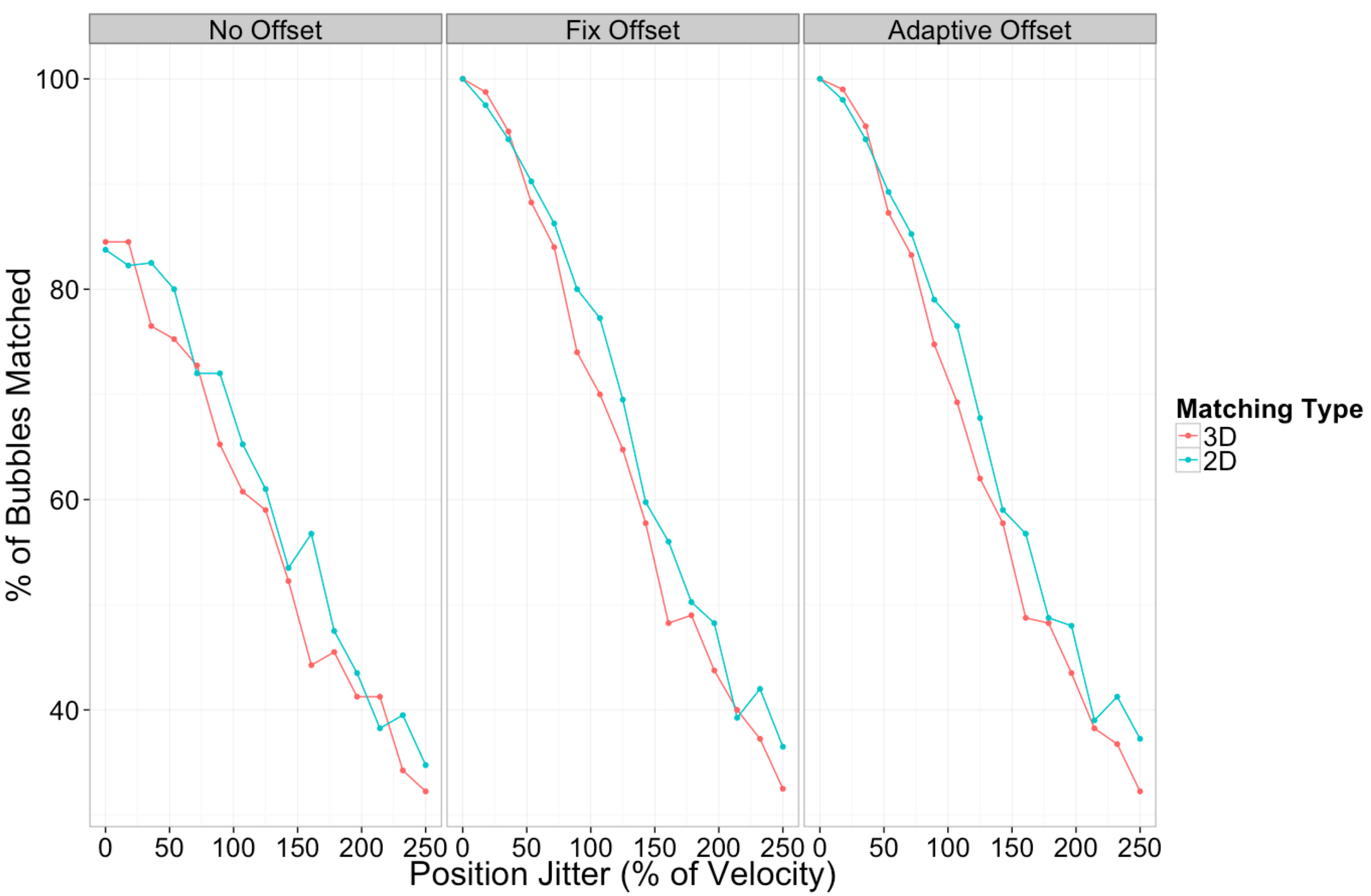


+/- R Code

Time	NN	ONN	ANN
1.9	22.5%	22.5%	25%
3.7	22.5%	25%	25%
5.5	32.5%	35%	32.5%

7.3	32.5%	32.5%	27.5%
9.1	20%	30%	22.5%
10.9	20%	20%	25%
12.7	12.5%	17.5%	17.5%
14.5	17.5%	25%	20%
16.3	15%	17.5%	12.5%
18.1	22.5%	27.5%	25%

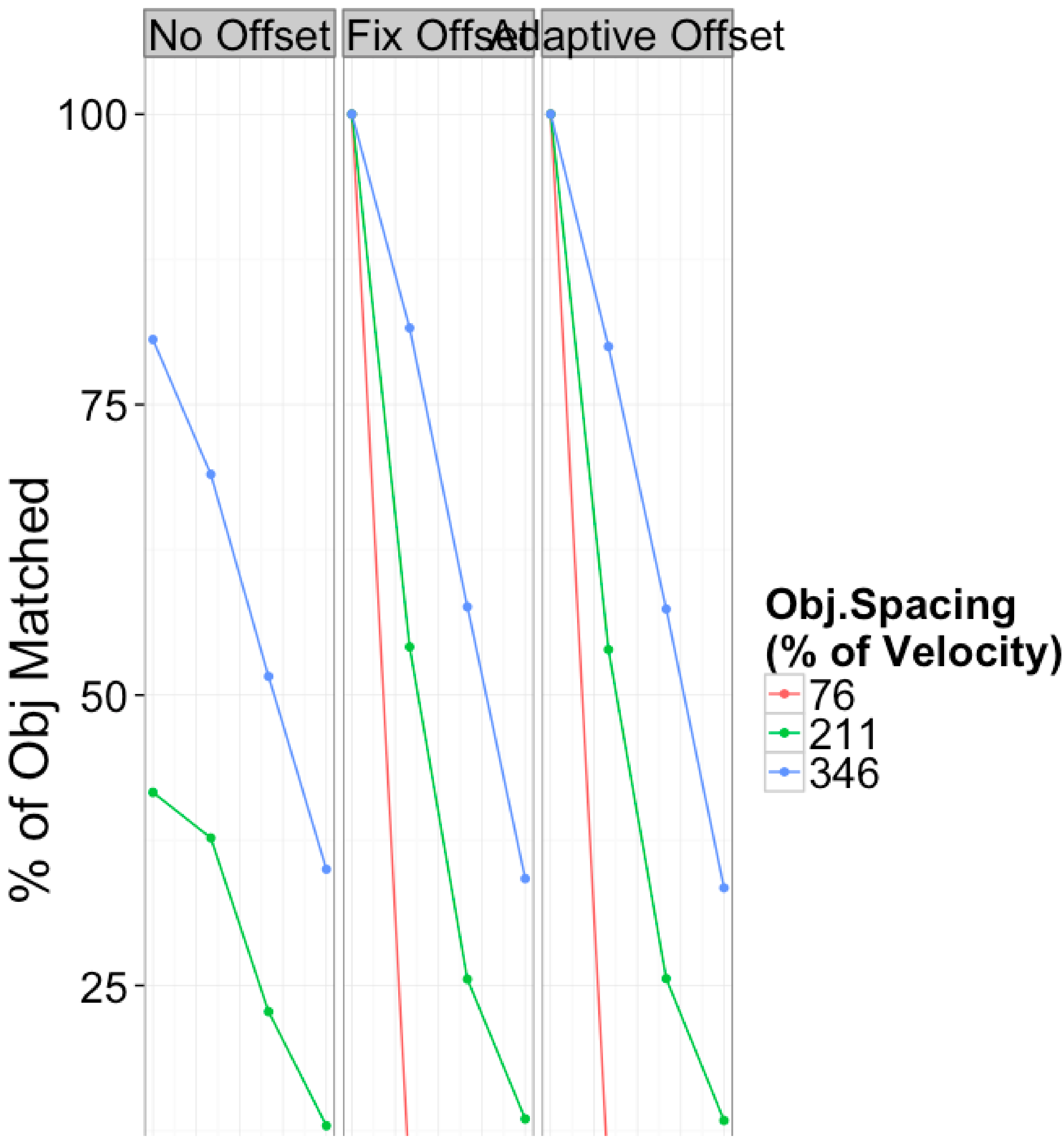
Jitter Sensitivity

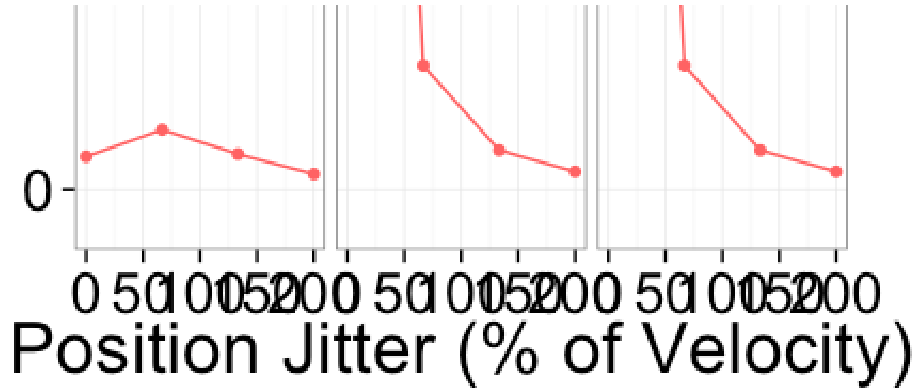


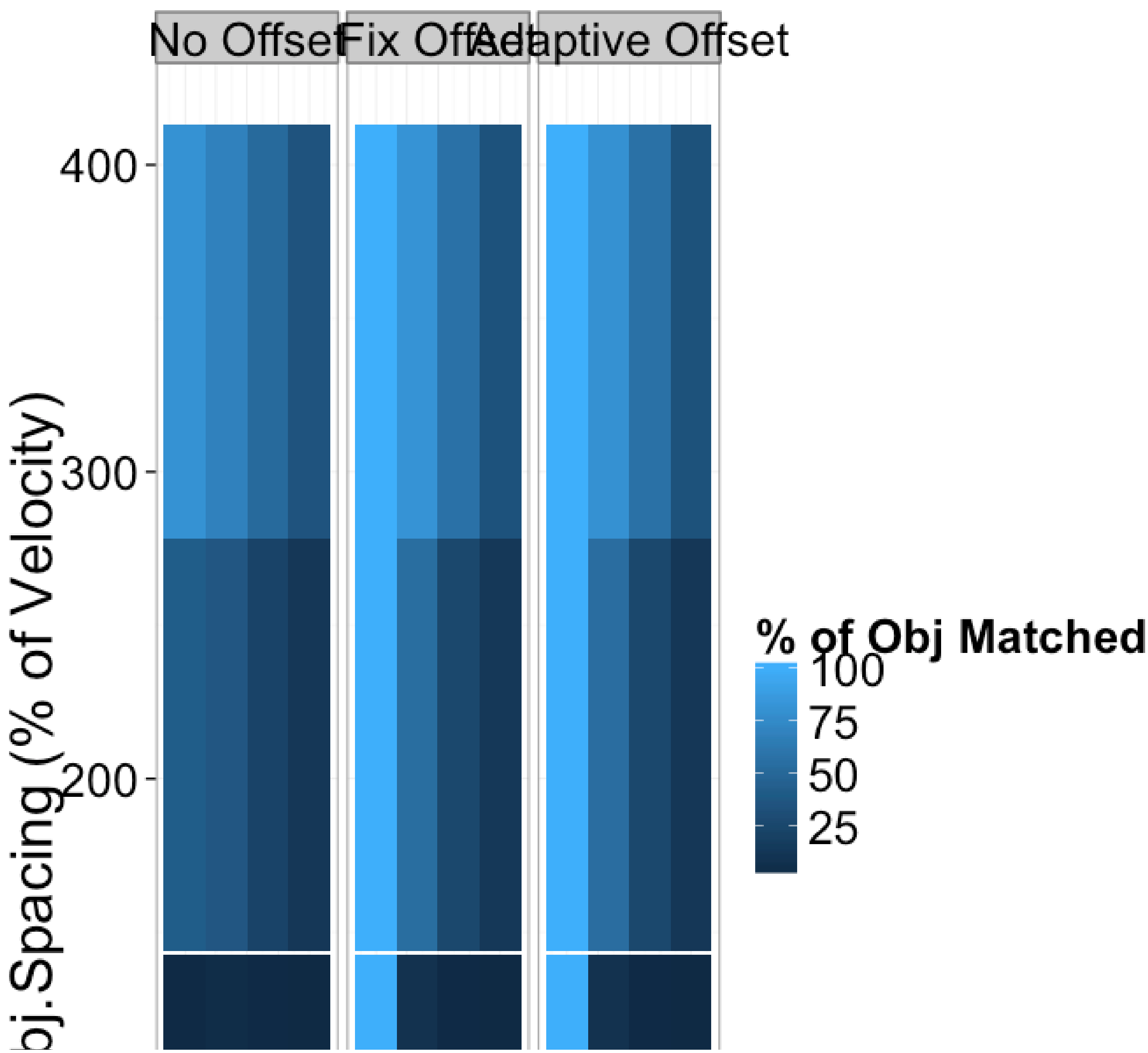
+/- R Code

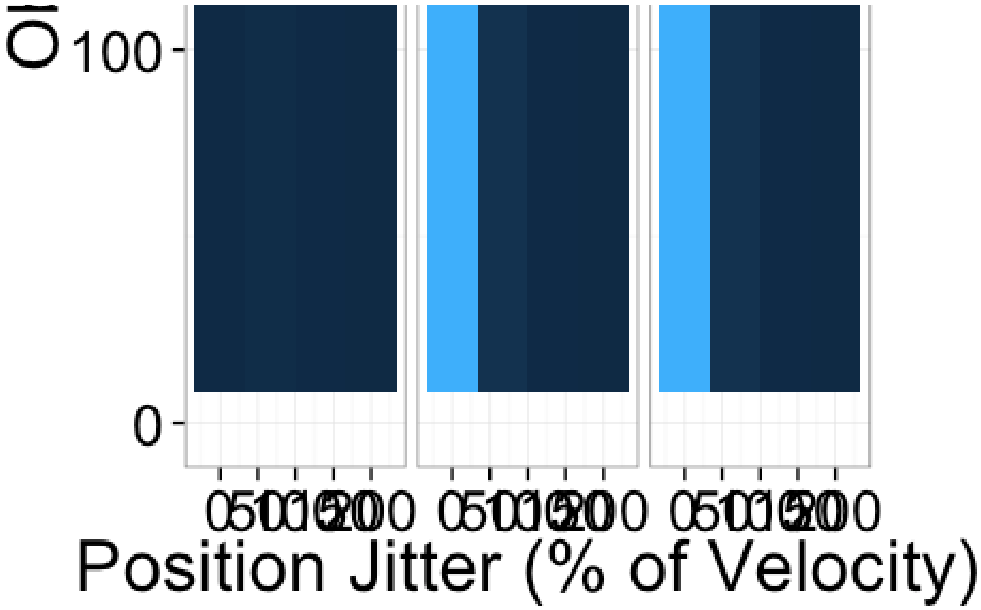
Density and Jitter

+/- R Code




0
0 50 100 150 200 0 50 100 150 200 0 50 100 150 200
Position Jitter (% of Velocity)


No Offset
Fix Offset
Adaptive Offset
400
300
200
Obj.Spacing (% of Velocity)
% of Obj Matched
100
75
50
25



Specifications

+/- R Code

jitter	mean_obj_spacing	Matching	obj.count	Matched	obj.matched	obj.found
0.0000	0.7563	No Offset	2500.0	2.156	1078	2.156
0.0000	0.7563	Fix Offset	2500.0	100.000	50000	100.000
0.0000	0.7563	Adaptive Offset	2500.0	100.000	50000	100.000
0.0000	2.1113	No Offset	111.4	41.622	924	41.478
0.0000	2.1113	Fix Offset	111.4	100.000	2220	99.655
0.0000	2.1113	Adaptive Offset	111.4	100.000	2220	99.655
0.0000	3.4568	No Offset	25.0	80.600	403	80.600
0.0000	3.4568	Fix Offset	25.0	100.000	500	100.000
0.0000	3.4568	Adaptive Offset	25.0	100.000	500	100.000
0.6667	0.7563	No Offset	2500.0	3.922	1961	3.922
0.6667	0.7563	Fix Offset	2500.0	8.122	4061	8.122
0.6667	0.7563	Adaptive Offset	2500.0	8.132	4066	8.132

0.6667	2.1113	No Offset	111.4	37.703	837	37.572
0.6667	2.1113	Fix Offset	111.4	54.144	1202	53.957
0.6667	2.1113	Adaptive Offset	111.4	53.919	1197	53.733
0.6667	3.4568	No Offset	25.0	69.000	345	69.000
0.6667	3.4568	Fix Offset	25.0	81.600	408	81.600
0.6667	3.4568	Adaptive Offset	25.0	80.000	400	80.000
1.3333	0.7563	No Offset	2500.0	2.326	1163	2.326
1.3333	0.7563	Fix Offset	2500.0	2.580	1290	2.580
1.3333	0.7563	Adaptive Offset	2500.0	2.574	1287	2.574
1.3333	2.1113	No Offset	111.4	22.748	505	22.669
1.3333	2.1113	Fix Offset	111.4	25.541	567	25.452
1.3333	2.1113	Adaptive Offset	111.4	25.586	568	25.497
1.3333	3.4568	No Offset	25.0	51.600	258	51.600
1.3333	3.4568	Fix Offset	25.0	57.600	288	57.600
1.3333	3.4568	Adaptive Offset	25.0	57.400	287	57.400
2.0000	0.7563	No Offset	2500.0	1.010	505	1.010
2.0000	0.7563	Fix Offset	2500.0	1.168	584	1.168
2.0000	0.7563	Adaptive Offset	2500.0	1.166	583	1.166
2.0000	2.1113	No Offset	111.4	12.928	287	12.883
2.0000	2.1113	Fix Offset	111.4	13.514	300	13.467
2.0000	2.1113	Adaptive Offset	111.4	13.378	297	13.332
2.0000	3.4568	No Offset	25.0	35.000	175	35.000
2.0000	3.4568	Fix Offset	25.0	34.200	171	34.200
2.0000	3.4568	Adaptive Offset	25.0	33.400	167	33.400

Designing Experiments

How does this help us to design experiments?

- density can be changed by adjusting the concentration of the substances being examined or the field of view
- flow per frame (image velocity) can usually be adjusted by changing pressure or acquisition time
- jitter can be estimated from images

How much is enough?

- difficult to create one number for every experiment
- 5% error in bubble position
  - → <5% in flow field
  - → >20% error in topology
- 5% error in shape or volume
  - → 5% in distribution or changes
  - → >5% in individual bubble changes
  - → >15% for single bubble strain tensor calculations

Extending Nearest Neighbor

Bijective Requirement

$$\vec{P}_f = \text{argmin}(\|\vec{P}_0 - \vec{y}\| \forall \vec{y} \in I_1)$$

$$\vec{P}_i = \text{argmin}(\|\vec{P}_f - \vec{y}\| \forall \vec{y} \in I_0)$$

$$\vec{P}_i \stackrel{?}{=} \vec{P}_0$$

Maximum Displacement

$$\vec{P}_1 = \begin{cases} \|\vec{P}_0 - \vec{y}\| < \text{MAXD}, & \text{argmin}(\|\vec{P}_0 - \vec{y}\| \forall \vec{y} \in I_1) \\ \text{Otherwise}, & \emptyset \end{cases}$$

Prior / Expected Movement

$$\vec{P}_1 = \text{argmin}(\|\vec{P}_0 + \vec{v}_{offset} - \vec{y}\| \forall \vec{y} \in I_1)$$

Adaptive Movement

Can then be calculated in an iterative fashion where the offset is the average from all of the $\vec{P}_1 - \vec{P}_0$ vectors. It can also be performed

$$\vec{P}_1 = \text{argmin}(||\vec{P}_0 + \vec{v}_{offset} - \vec{y}|| \forall \vec{y} \in I_1)$$

Beyond Nearest Neighbor

While nearest neighbor provides a useful starting tool it is not sufficient for truly complicated flows and datasets.

Better Approaches

1. Multiple Hypothesis Testing Nearest neighbor just compares the points between two frames and there is much more information available in most time-resolved datasets. This approach allows for multiple possible paths to be explored at the same time and the best chosen only after all frames have been examined

---

Shortcomings

1. Merging and Splitting Particles The simplicity of the nearest neighbor model does really allow for particles to merge and split (relaxing the bijective requirement allows such behavior, but the method is still not suited for such tracking). For such systems a more specific, physically-based is required to encapsulate this behavior.

Voxel-based Approaches

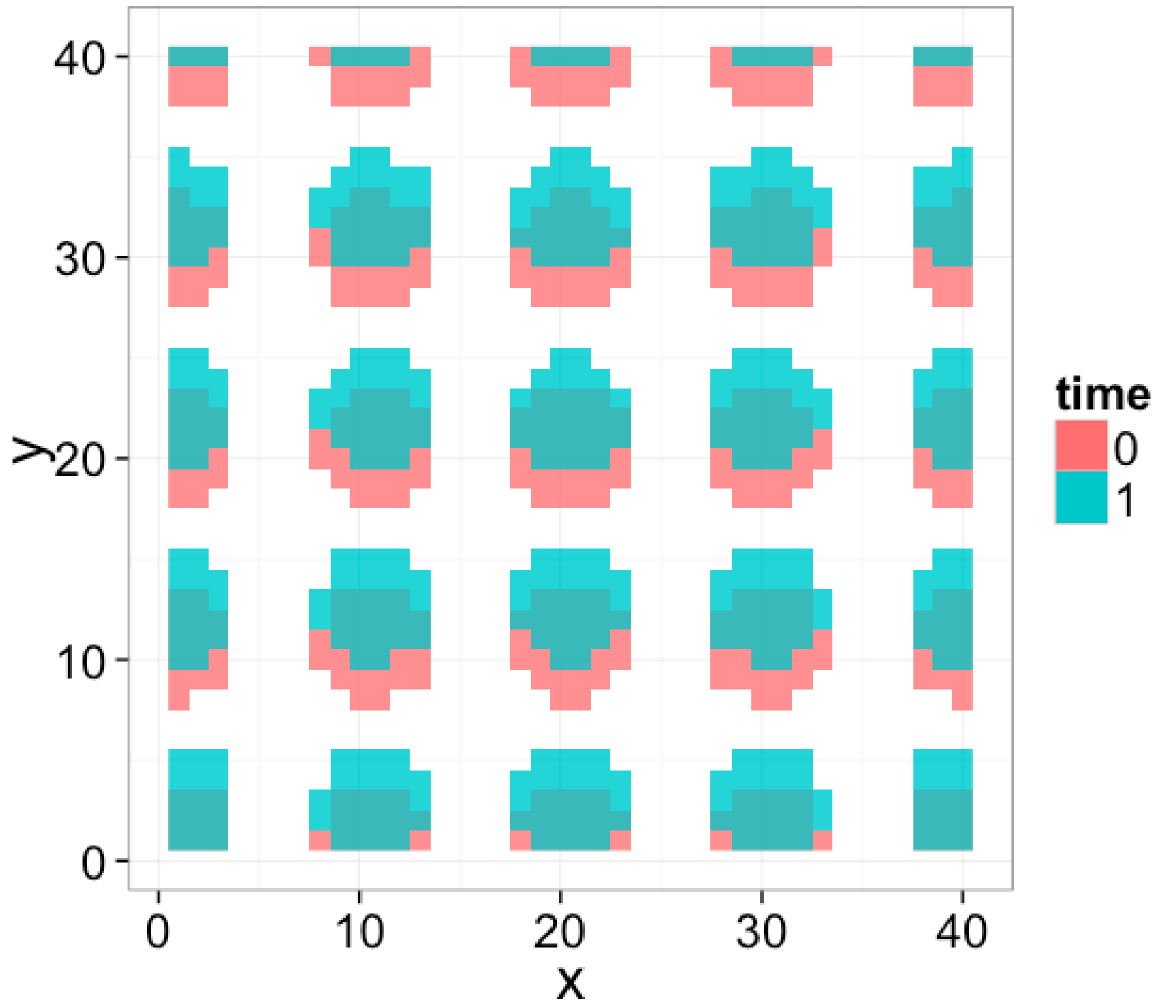
For voxel-based approachs the most common analyses are digital image correlation (or for 3D images digital volume correlation), where the correlation is calculated between two images or volumes.

Standard Image Correlation

Given images $I_0(\vec{x})$ and $I_1(\vec{x})$ at time t_0 and t_1 respectively. The correlation between these two images can be calculated

$$C_{I_0,I_1}(\vec{r}) = \langle I_0(\vec{x}) I_1(\vec{x} + \vec{r}) \rangle$$

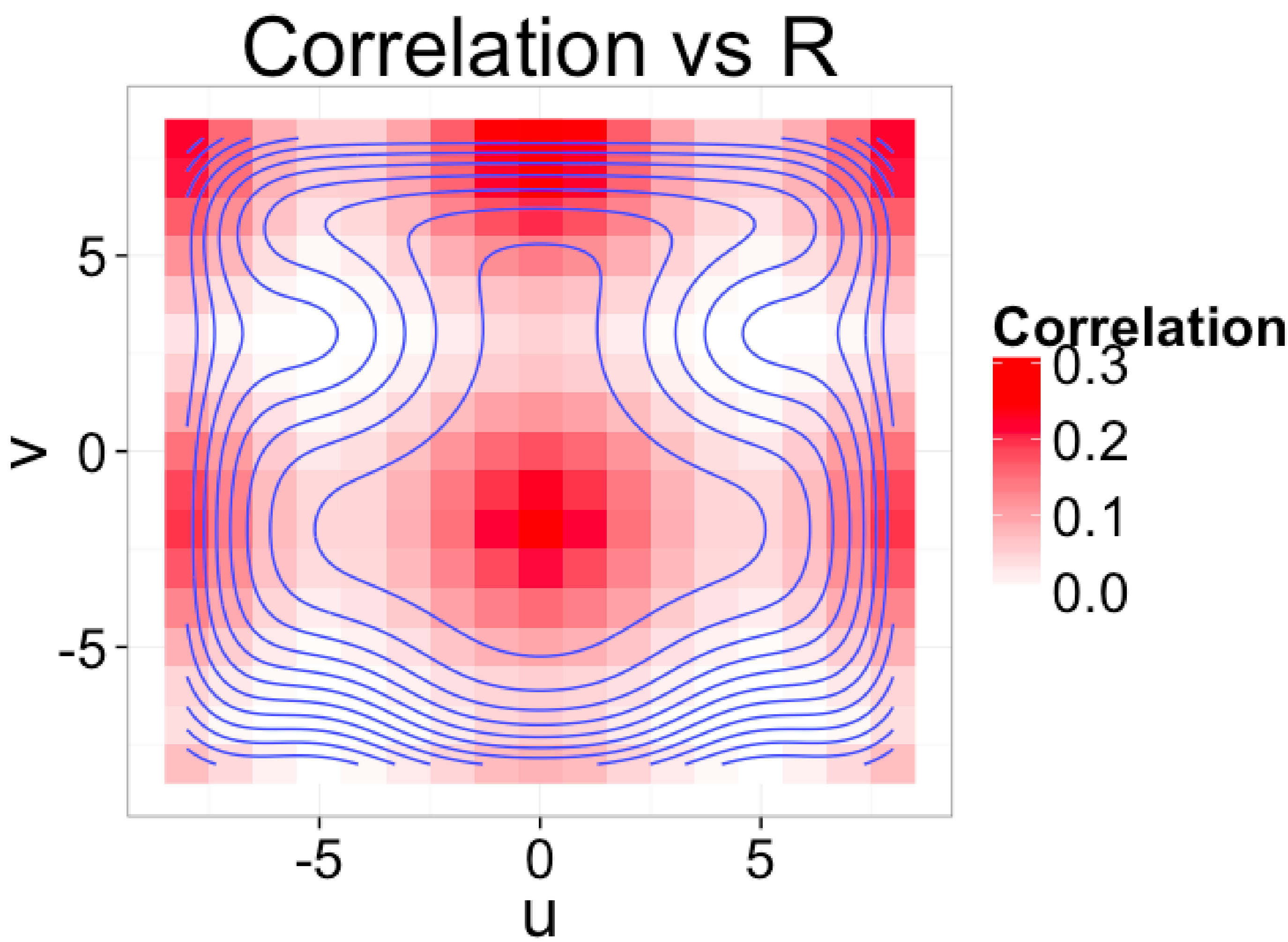
+/- R Code



+/- R Code

+/- R Code

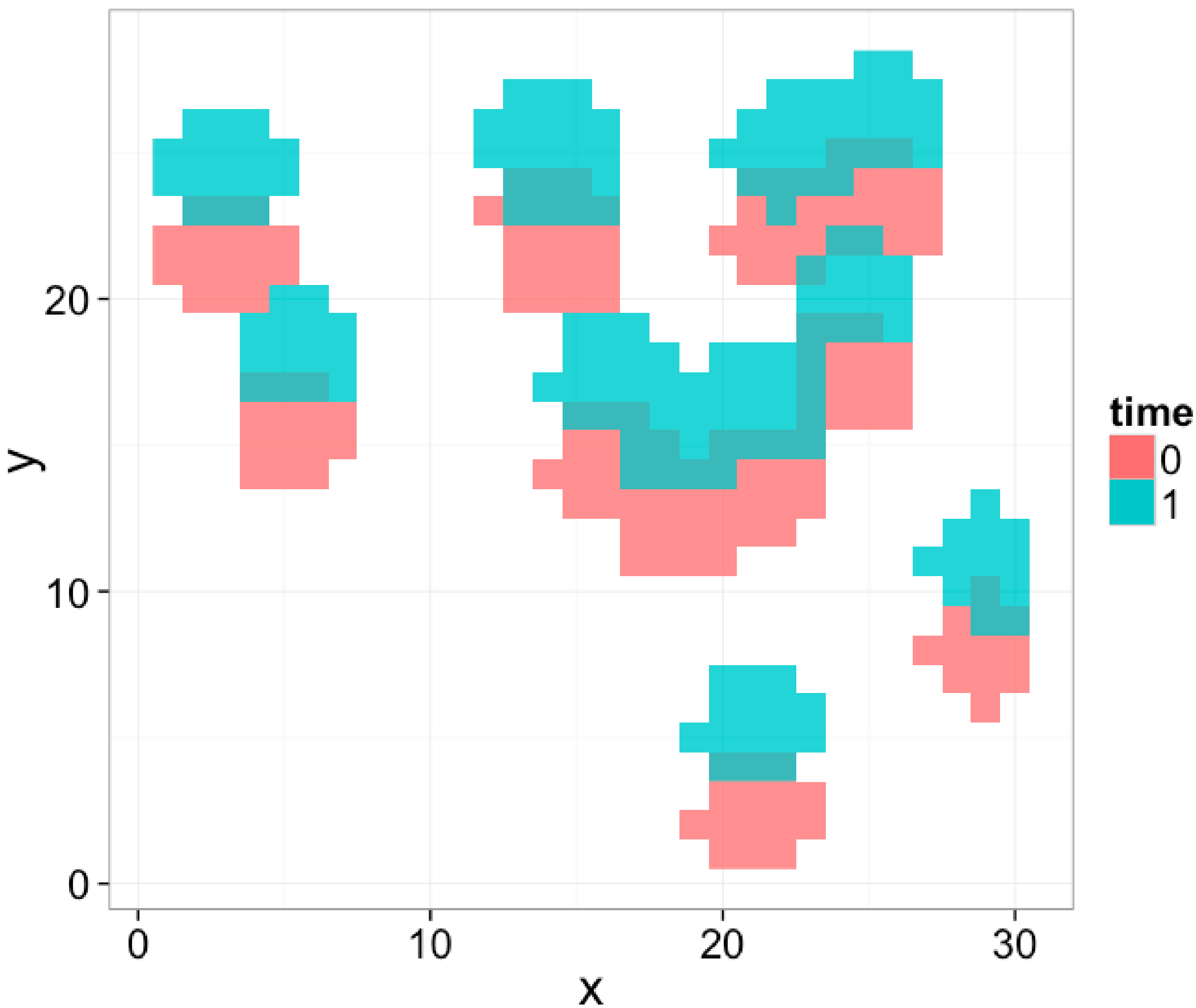
+/- R Code



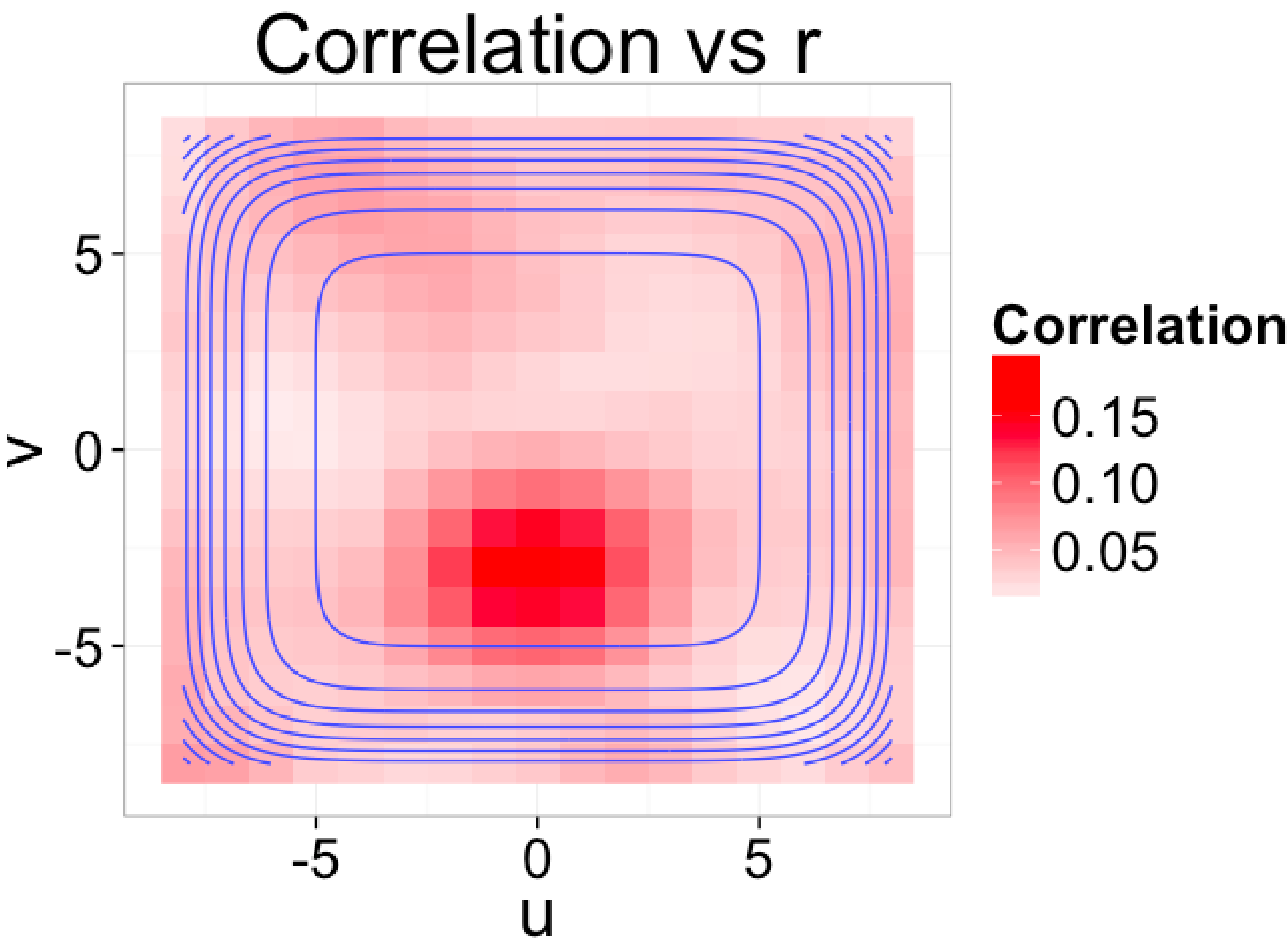
Random Image Positions

With highly structured / periodic samples identfying a best correlation is difficult since there are multiple maxima.

+/- R Code


y
x
0
10
20
30
0
10
20
time
0
1

+/- R Code

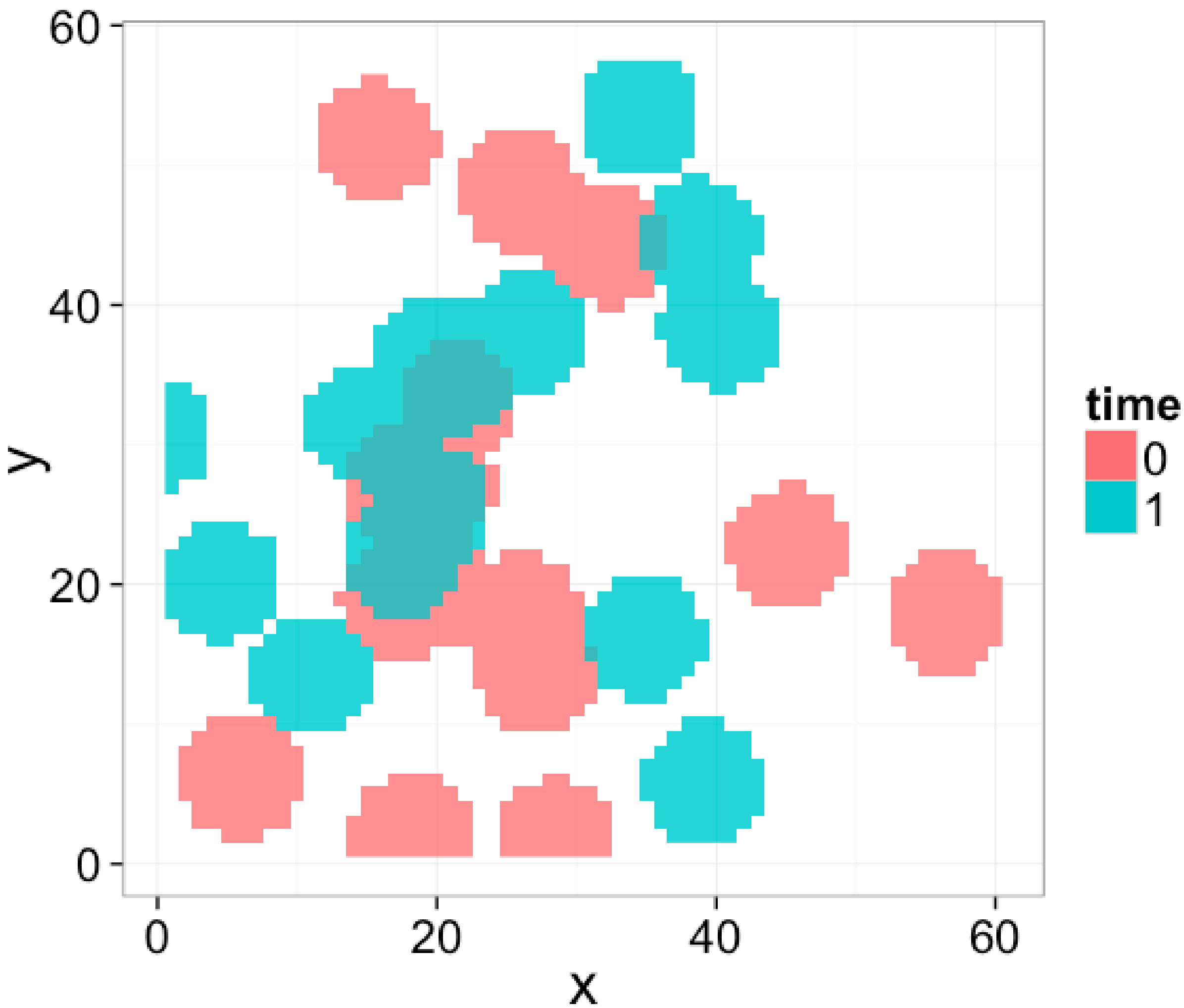


Extending Correlation

The correlation function can be extended by adding rotation and scaling terms to the offset making the tool more flexible but also more computationally expensive for large search spaces.

$$C_{I_0,I_1}(\vec{r}, s, \theta) =$$

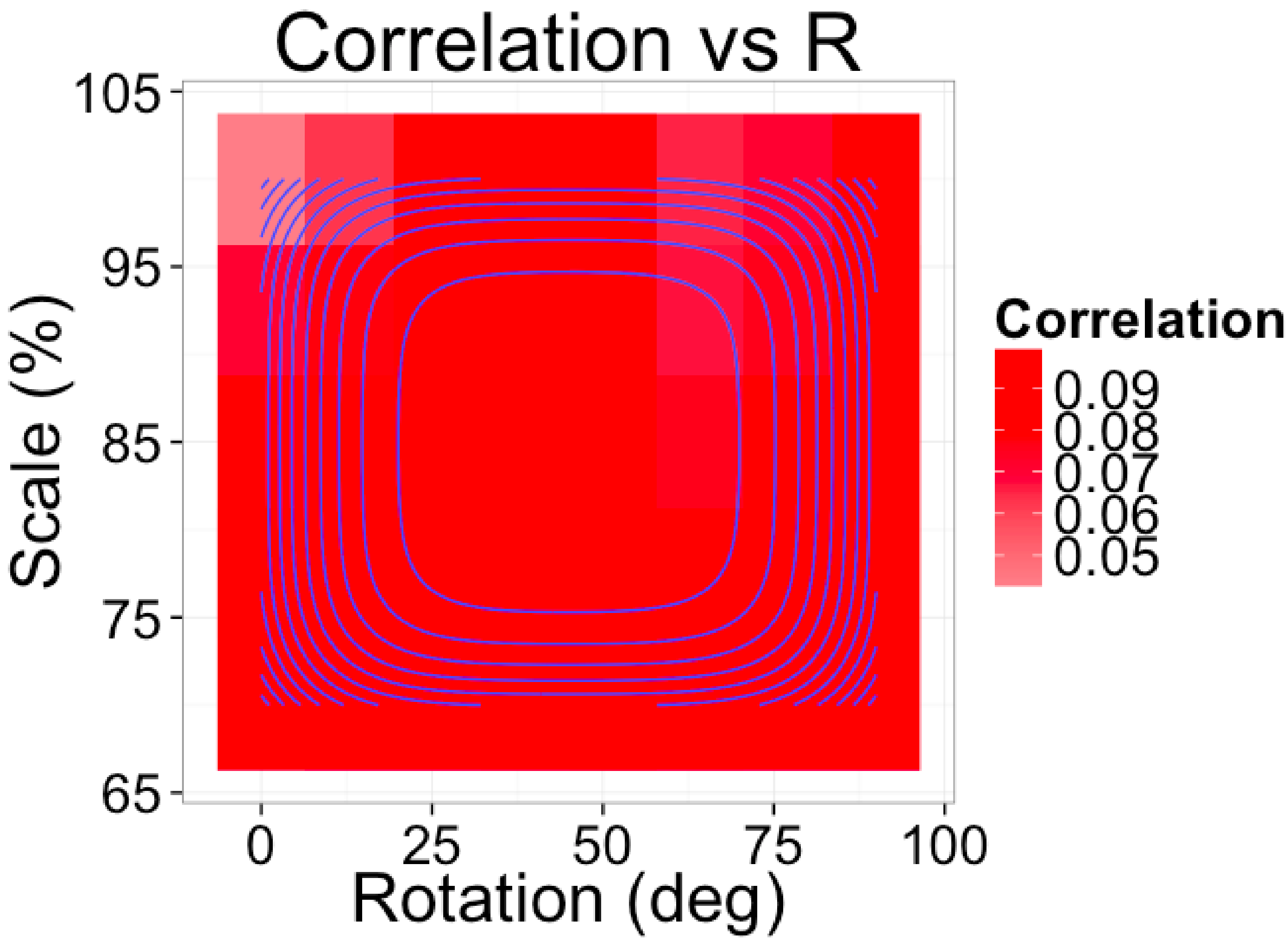
$$\langle I_0(\vec{x}) I_1(\begin{bmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{bmatrix} \vec{x} + \vec{r}) \rangle$$



+/- R Code

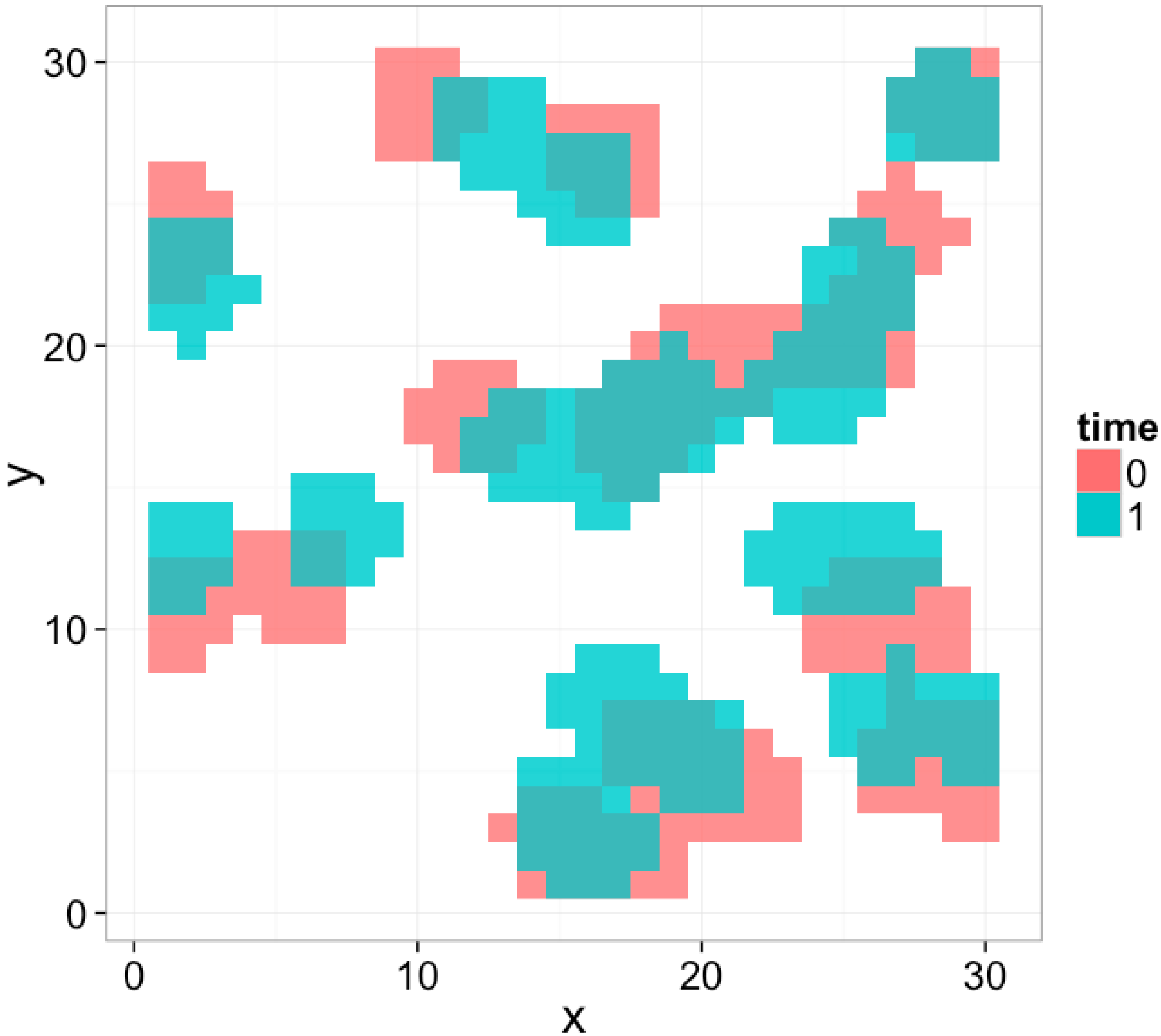
+/- R Code

+/- R Code

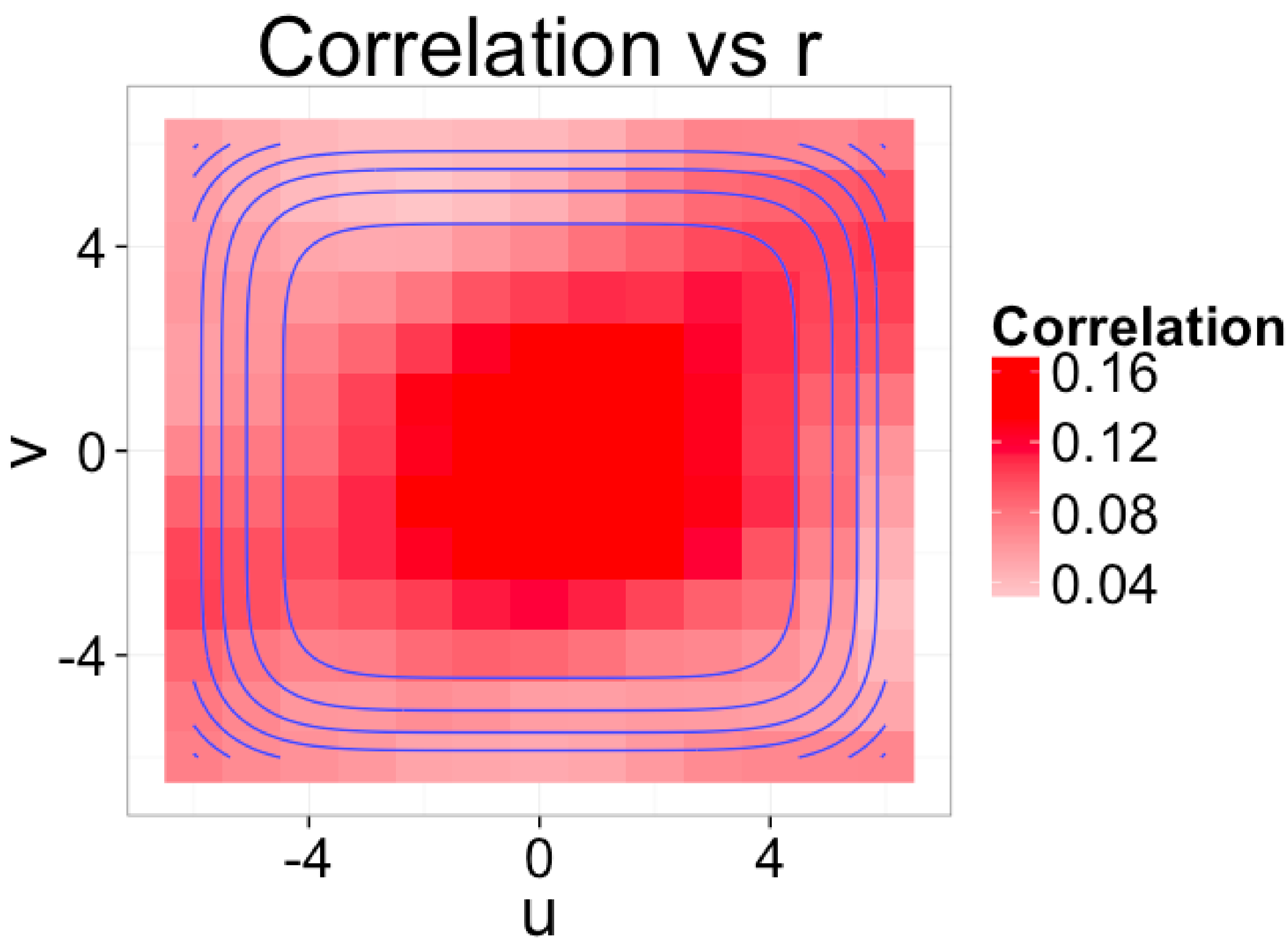


More Complicated Changes

+/- R Code



+/- R Code



+/- R Code

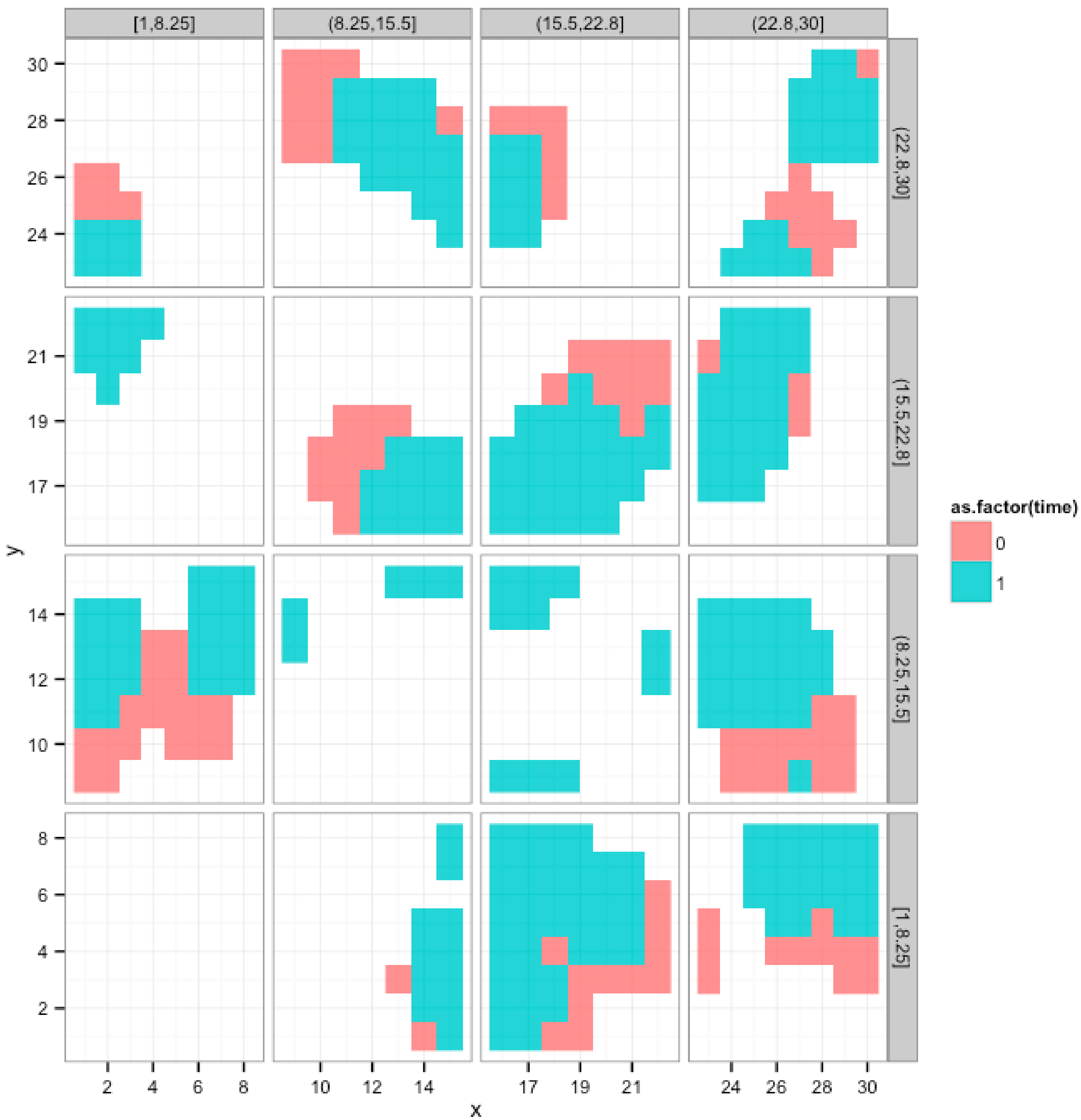
Subdividing the data

We can approach the problem by subdividing the data into smaller blocks and then apply the digital volume correlation independently to each block.

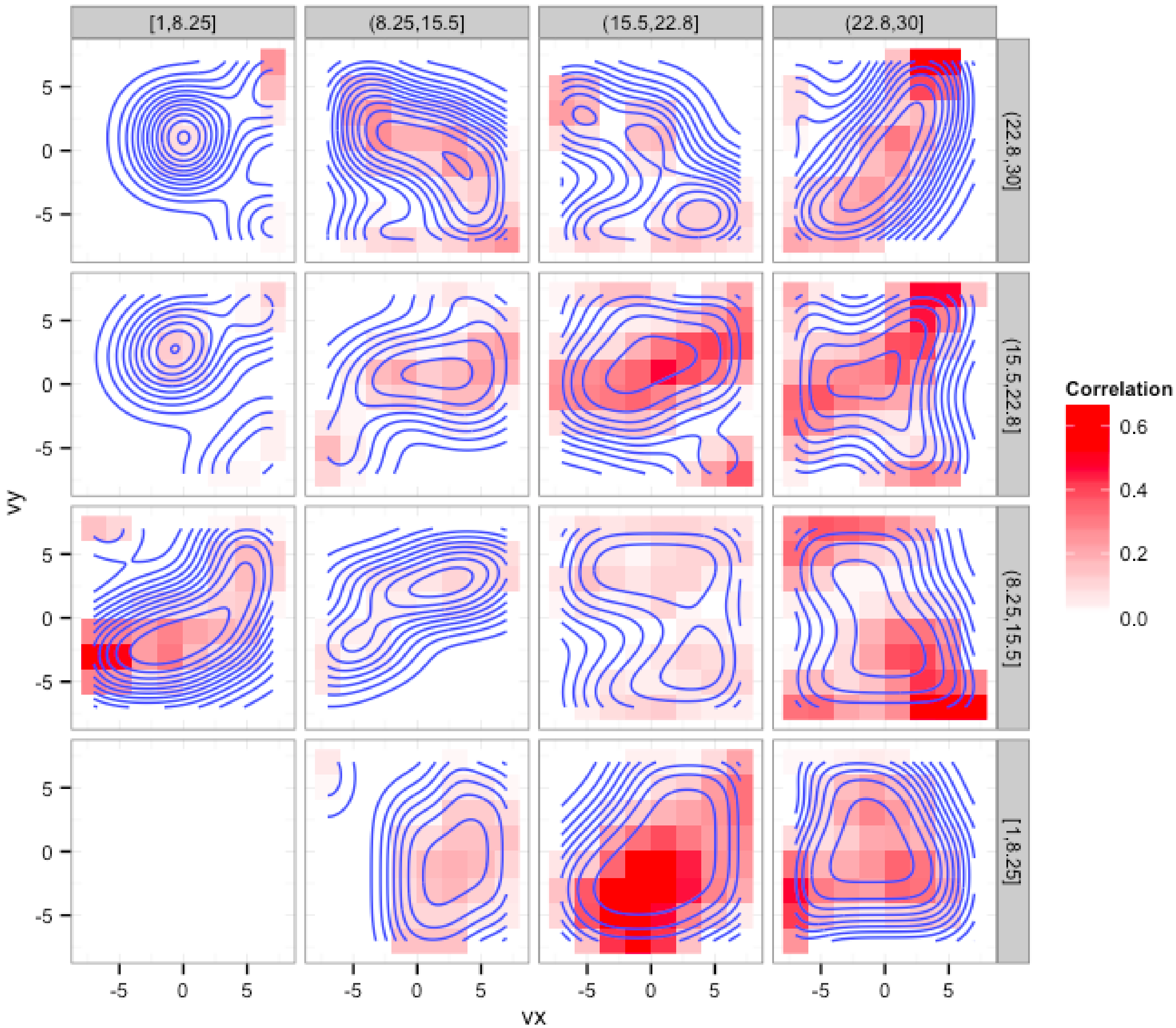
- information on changes in different regions
- less statistics than a larger box

Choosing a window size

+/- R Code

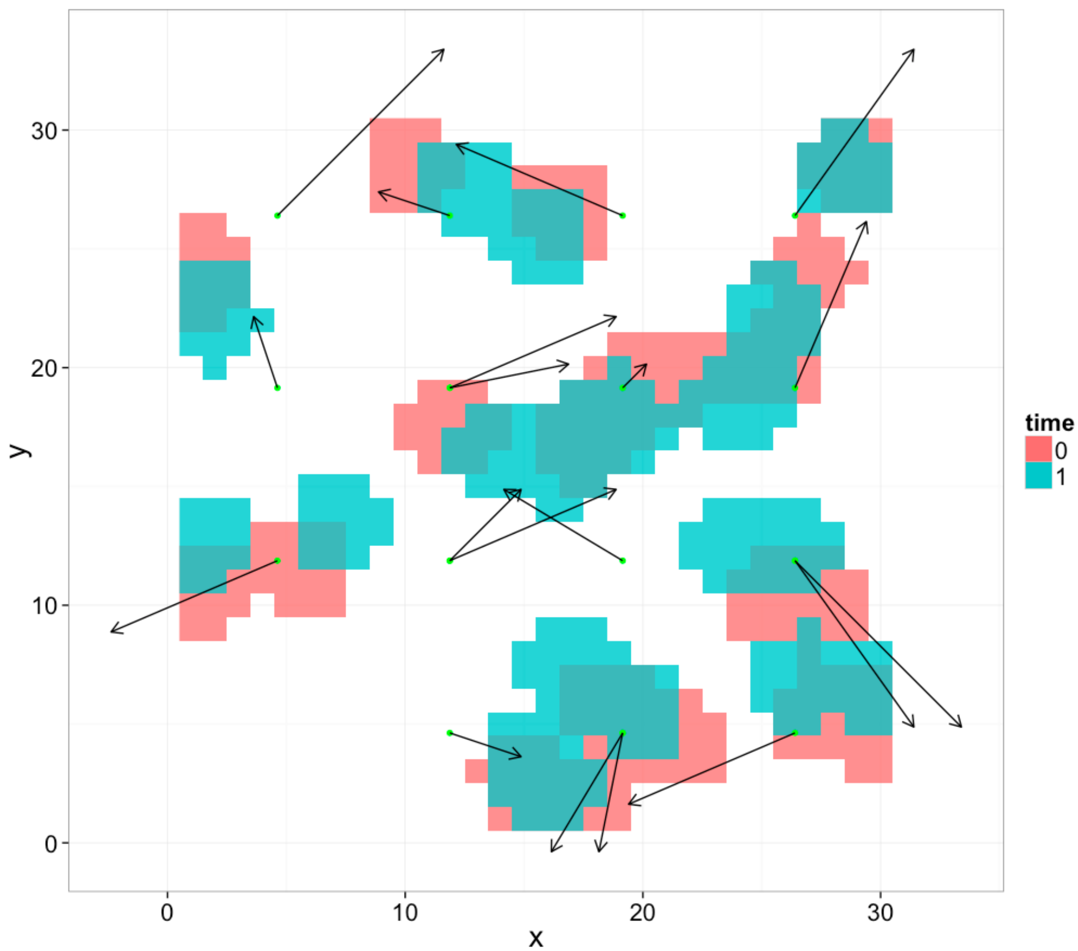

[1,8.25]
(8.25,15.5]
(15.5,22.8]
(22.8,30]
(22.8,30]
(15.5,22.8]
(8.25,15.5]
[1,8.25]
y
x
as.factor(time)
0
1

+/- R Code



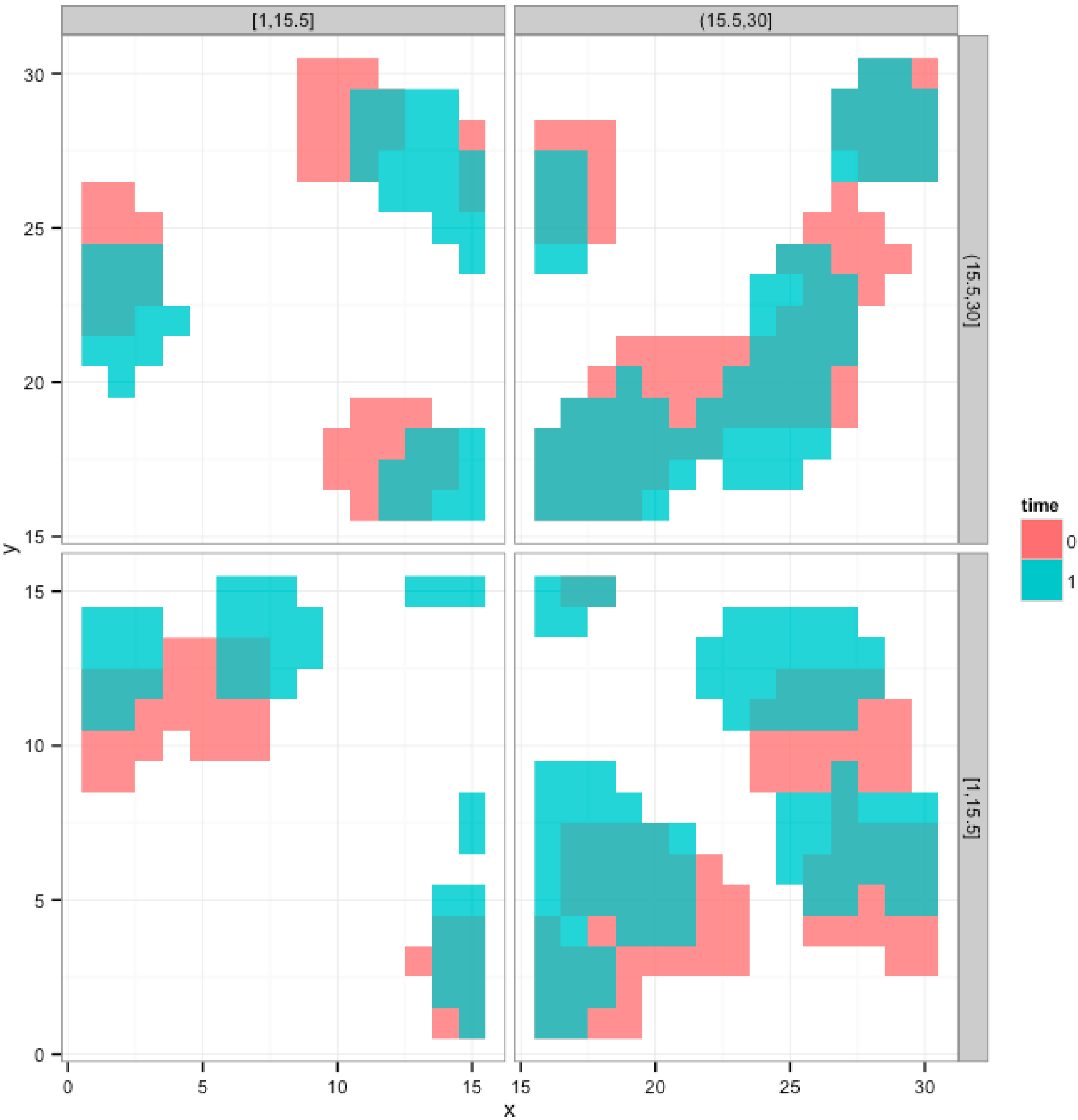
Overlap

+/- R Code

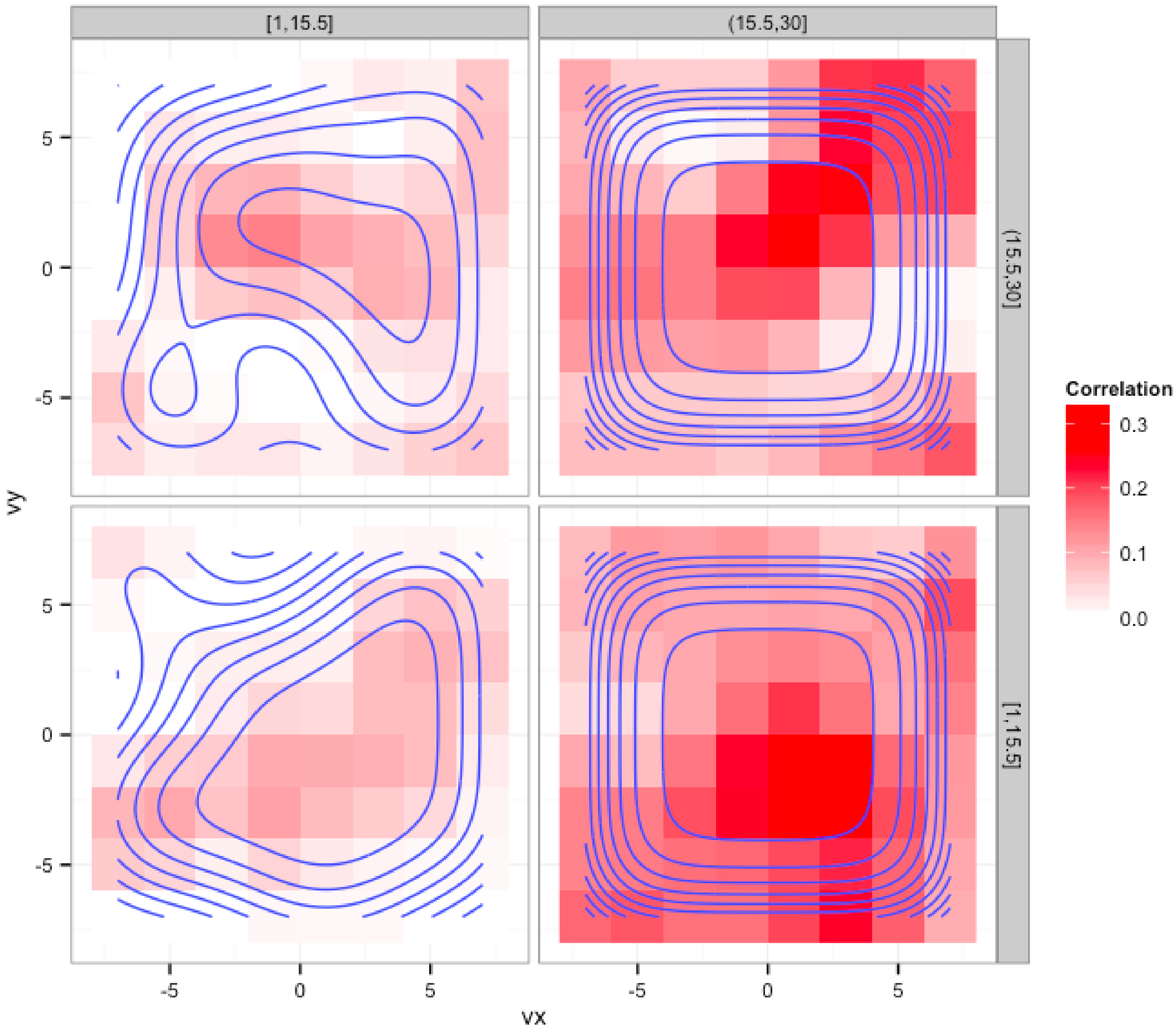


Too large of a window

+/- R Code

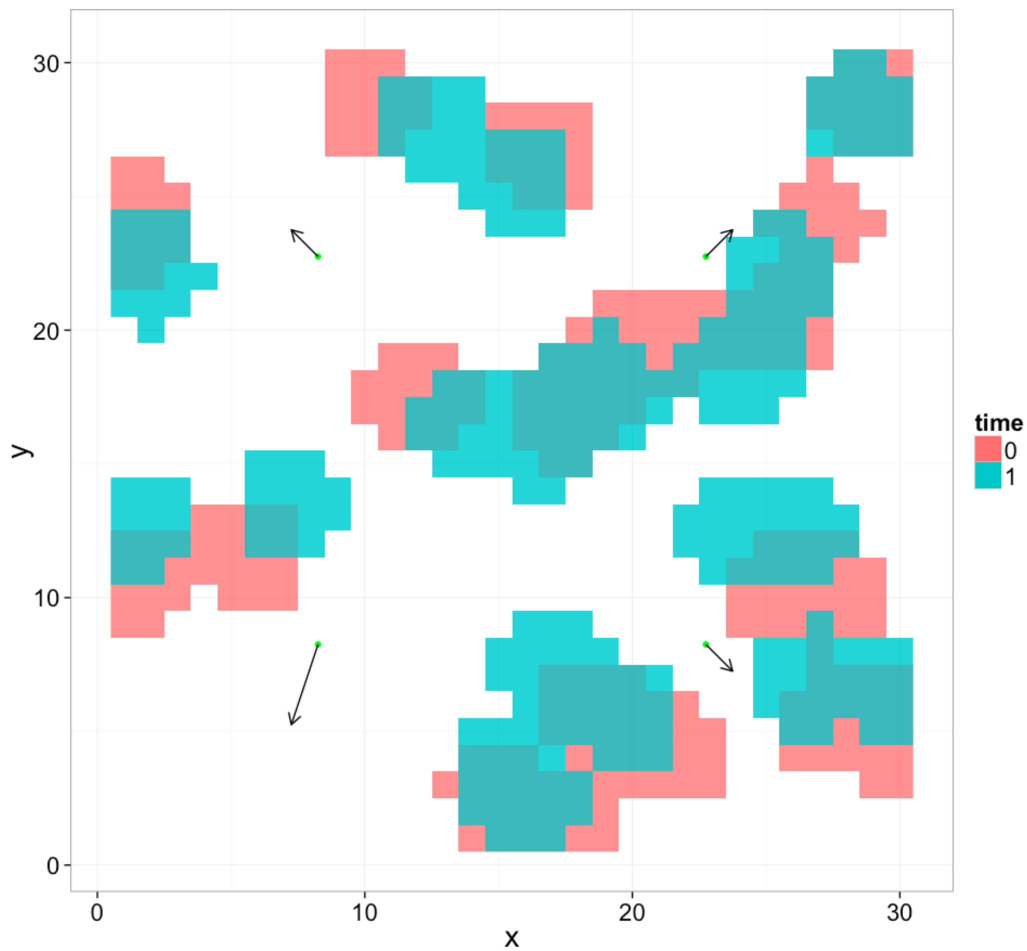


+/- R Code



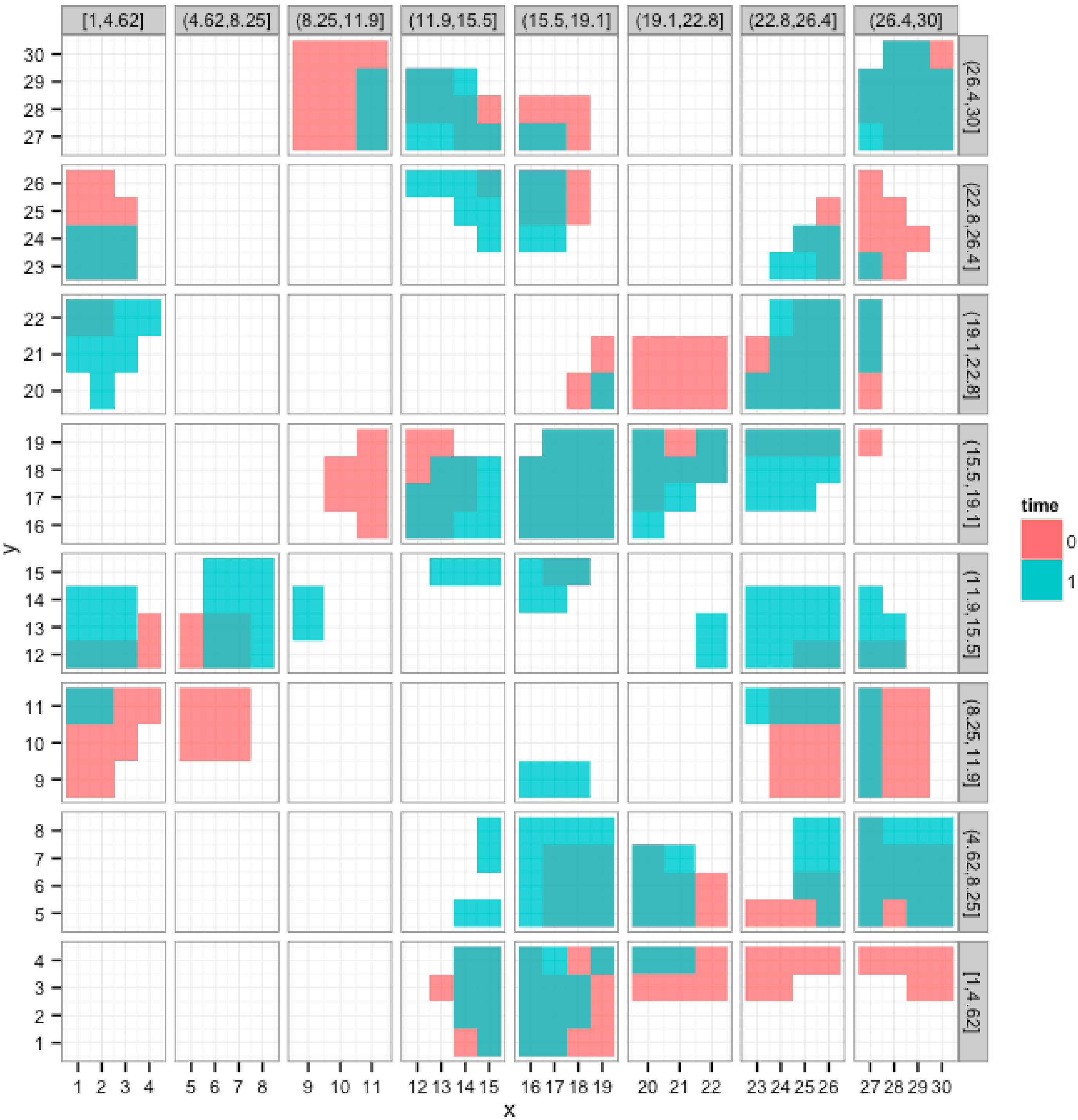
Overlap

+/- R Code

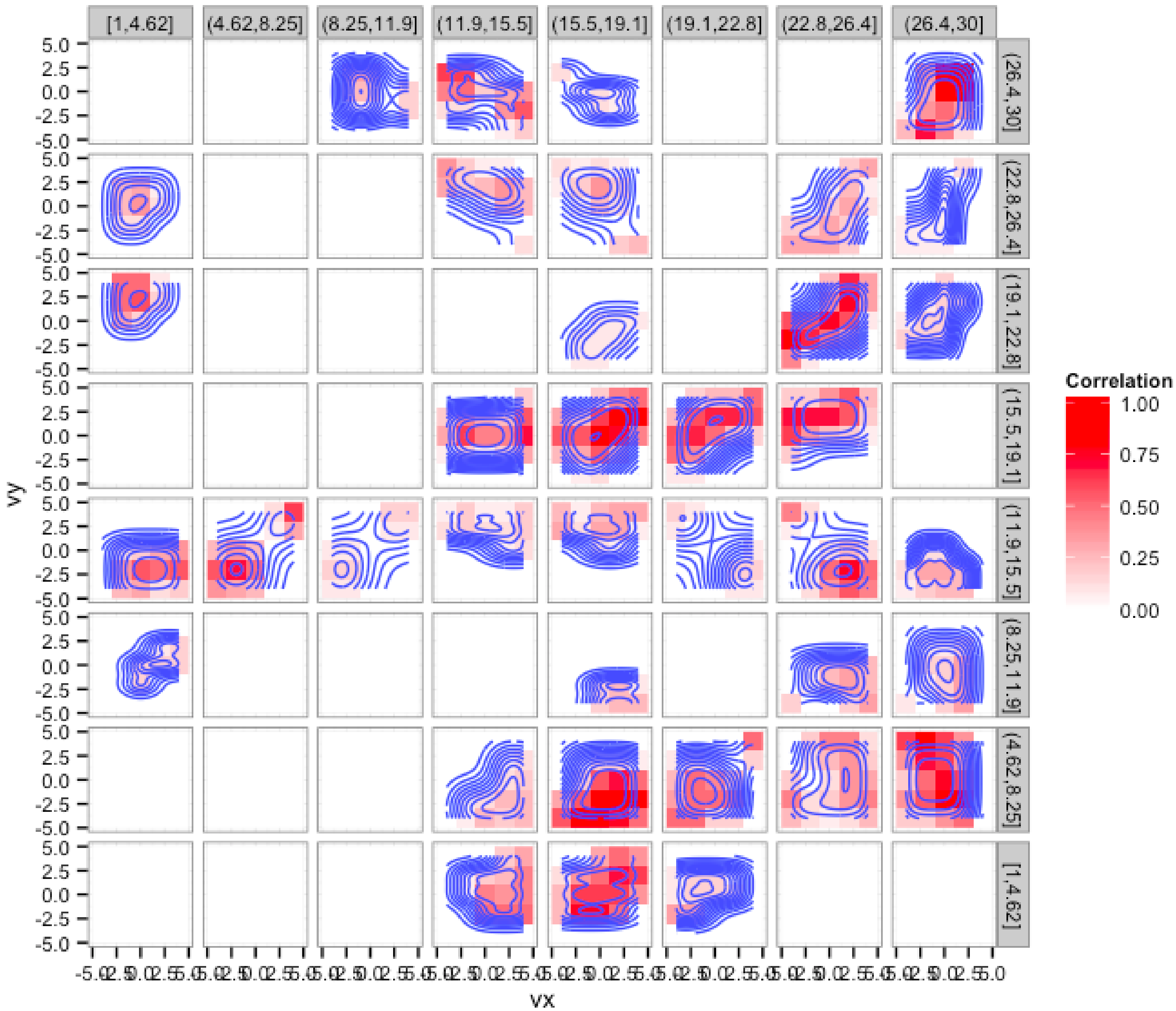


Too small of a window

+/- R Code

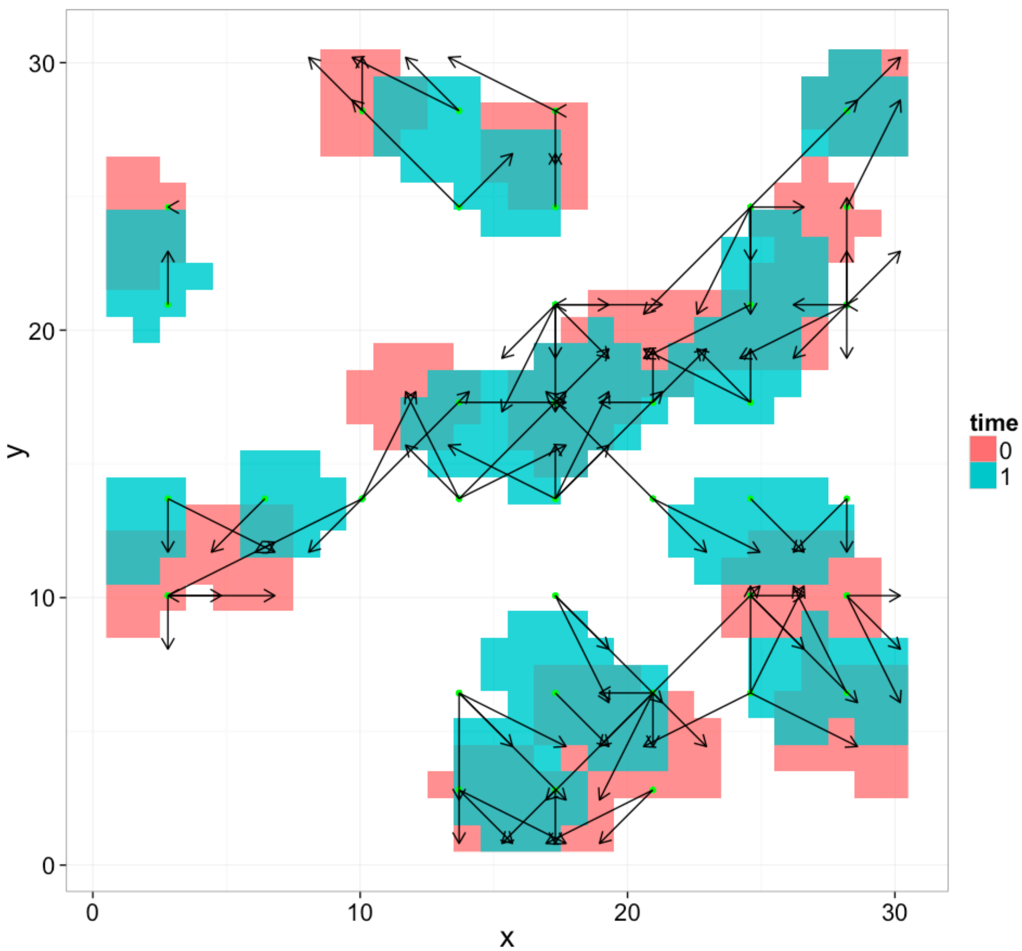

[1,4.62]
(4.62,8.25]
(8.25,11.9]
(11.9,15.5]
(15.5,19.1]
(19.1,22.8]
(22.8,26.4]
(26.4,30]
(26.4,30]
(22.8,26.4]
(19.1,22.8]
(15.5,19.1]
(11.9,15.5]
(8.25,11.9]
(4.62,8.25]
[1,4.62]
y
x
time
0
1

+/- R Code



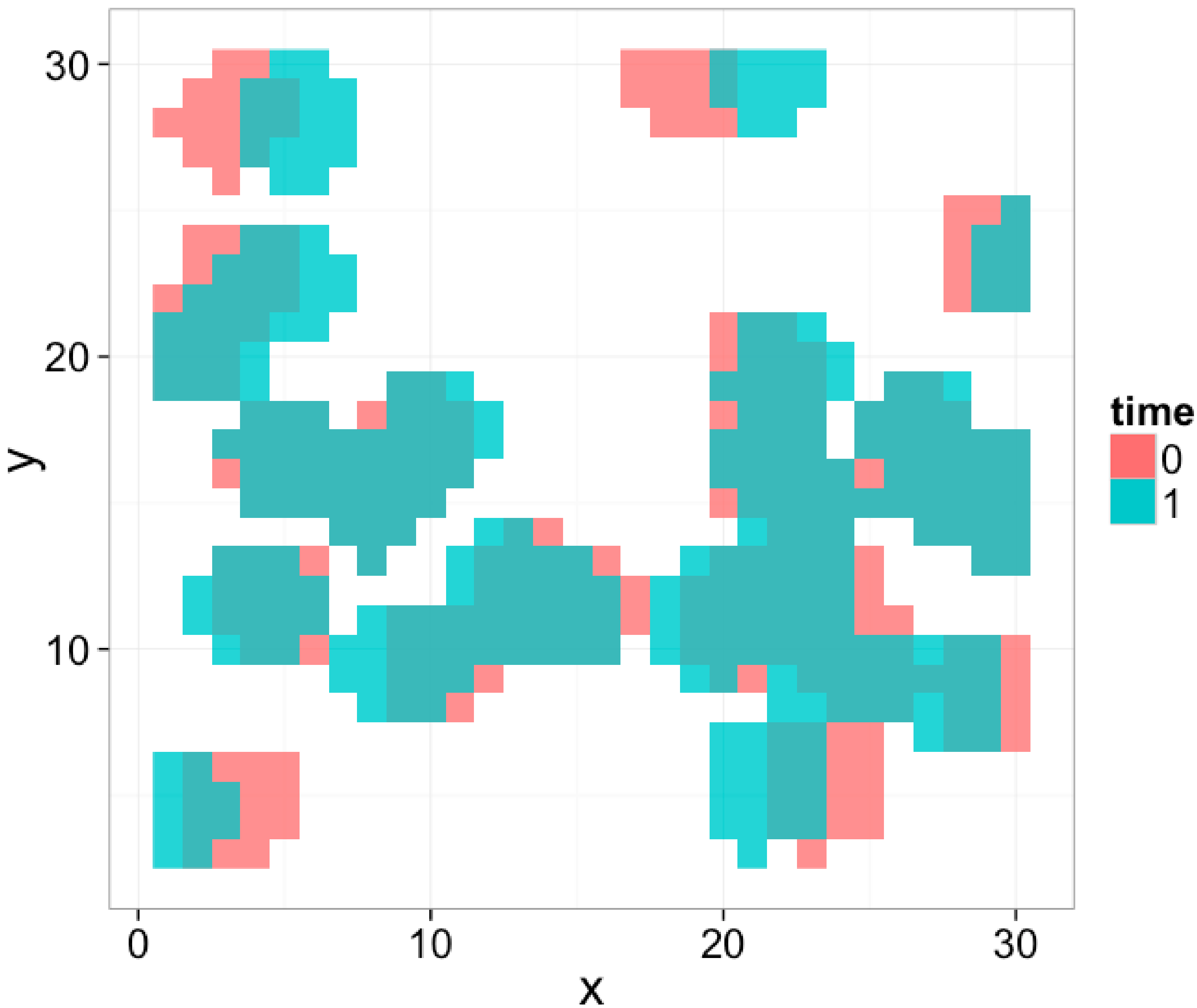
Overlap

+/- R Code

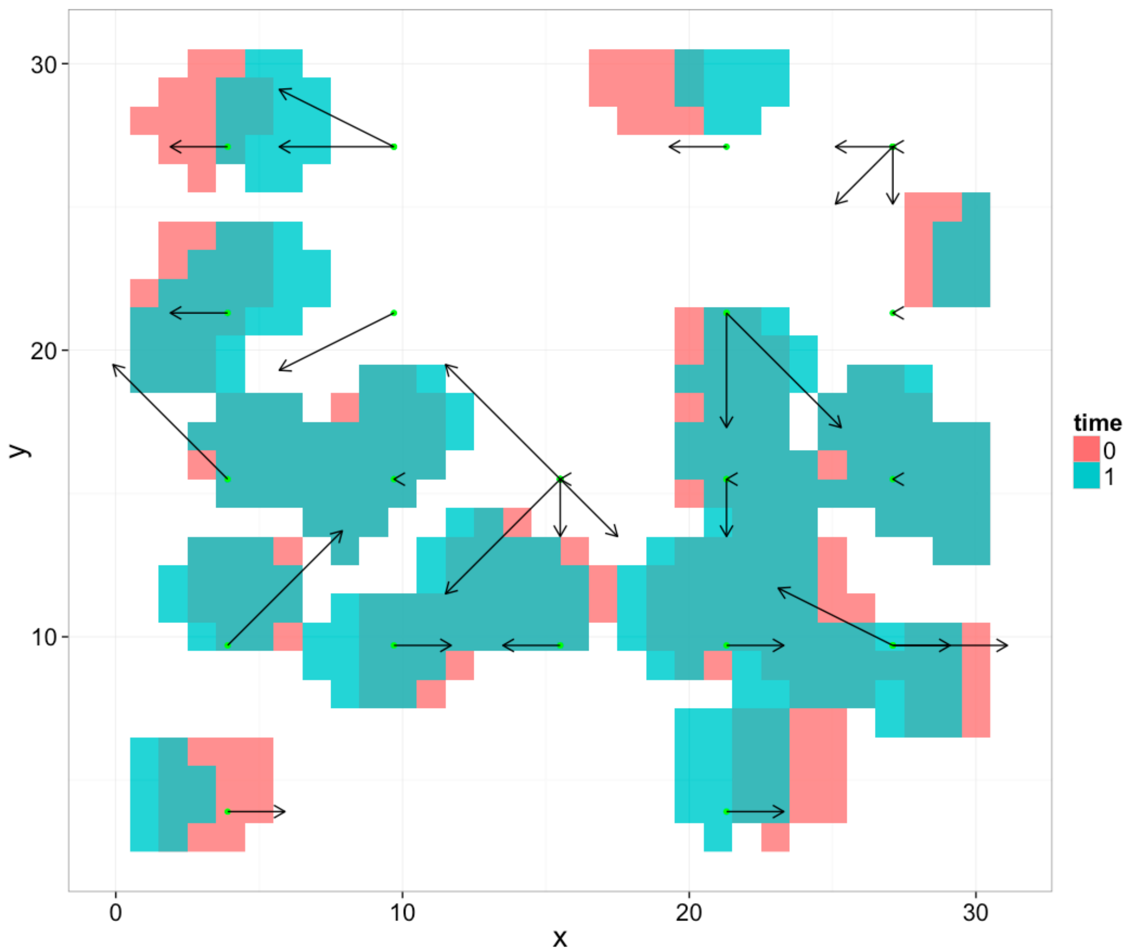


Shearing

+/- R Code

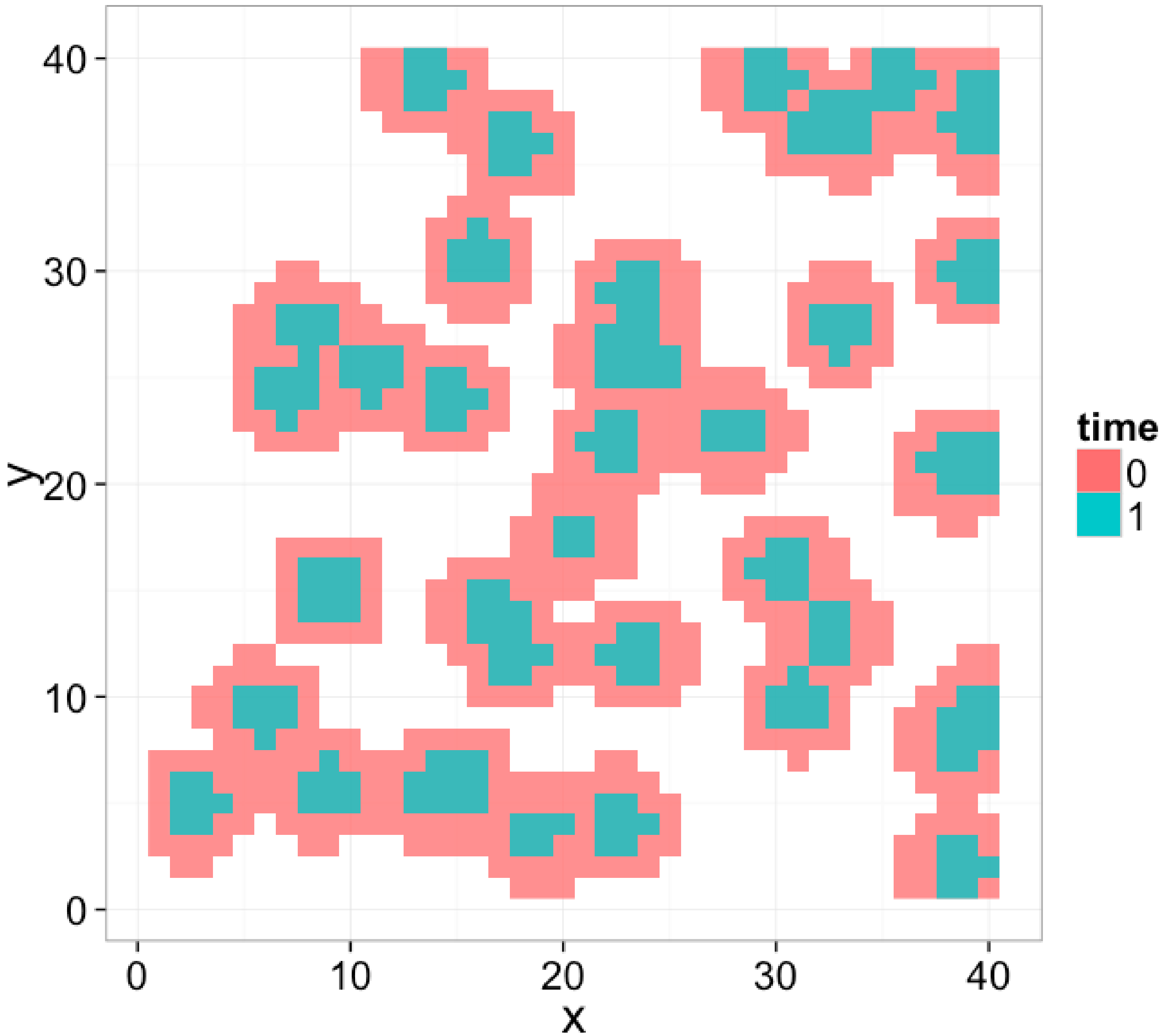


+/- R Code

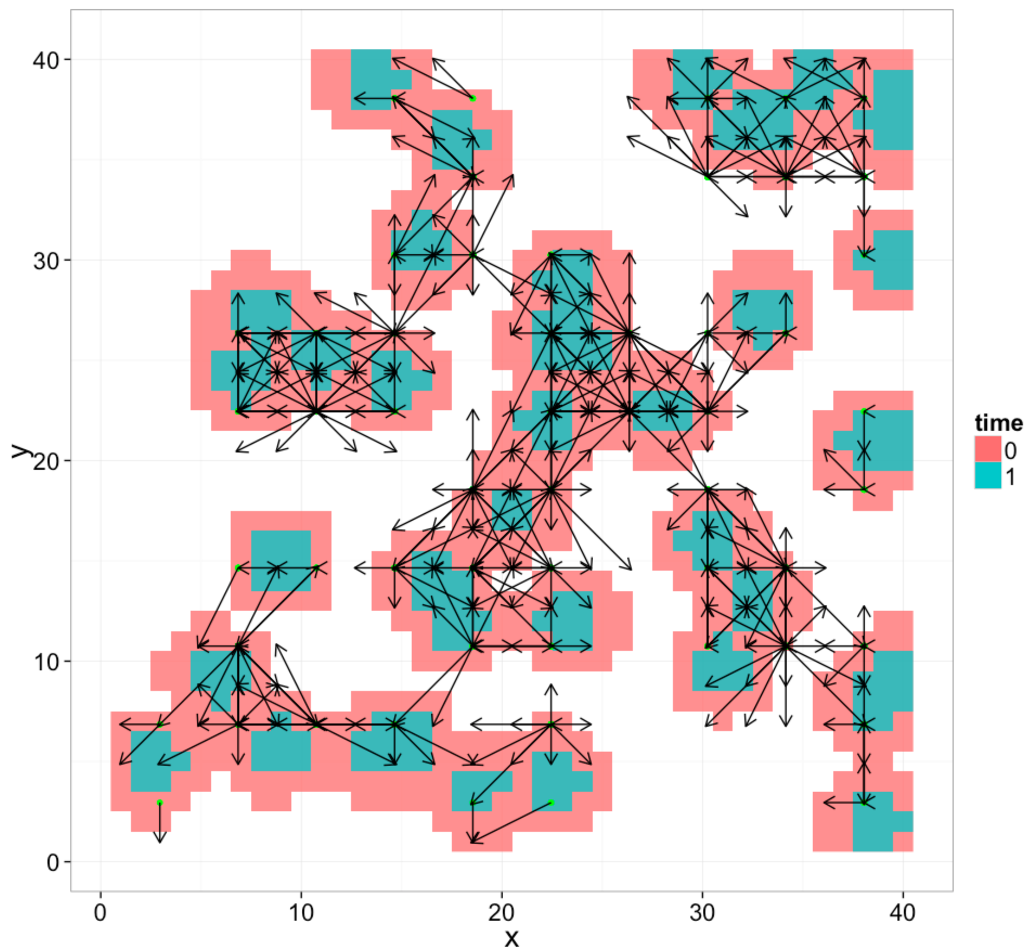


Compression

+/- R Code



+/- R Code



Distribution Metrics

As we covered before distribution metrics like the distribution tensor can be used for tracking changes inside a sample. Of these the most relevant is the texture tensor from cellular materials and liquid foam. The texture tensor is the same as the distribution tensor except that the edges (or faces) represent physically connected / touching objects rather than touching Voronoi faces (or conversely Delaunay triangles).

These metrics can also be used for tracking the behavior of a system without tracking the single points since most deformations of a system also deform the distribution tensor and can thus be extracted by comparing the distribution tensor at different time steps.

Quantifying Deformation: Strain

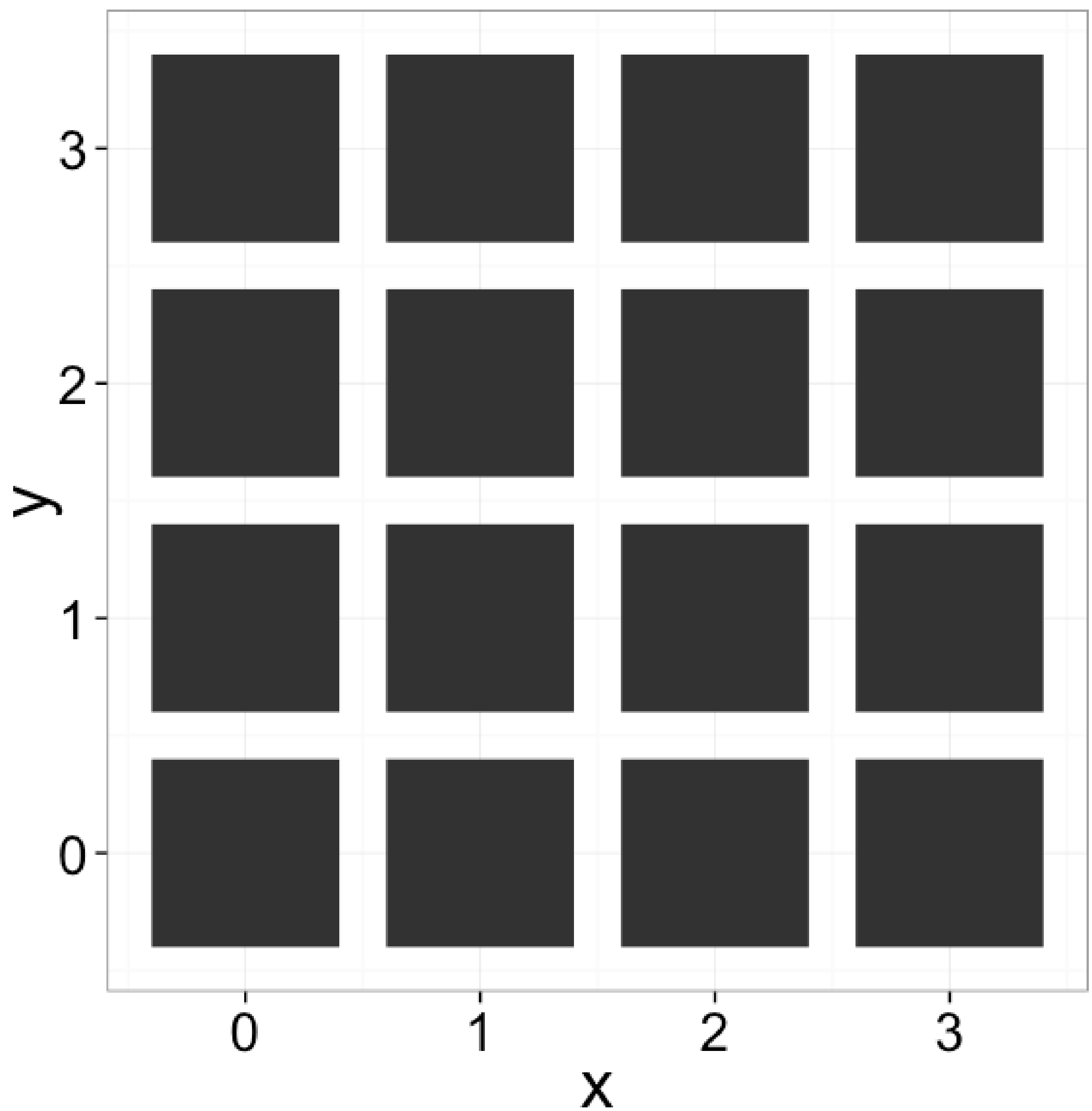
We can take any of these approaches and quantify the deformation using a tool called the strain tensor. Strain is defined in mechanics for the simple 1D case as the change in the length against the change in the original length.

$$e = \frac{\Delta L}{L}$$

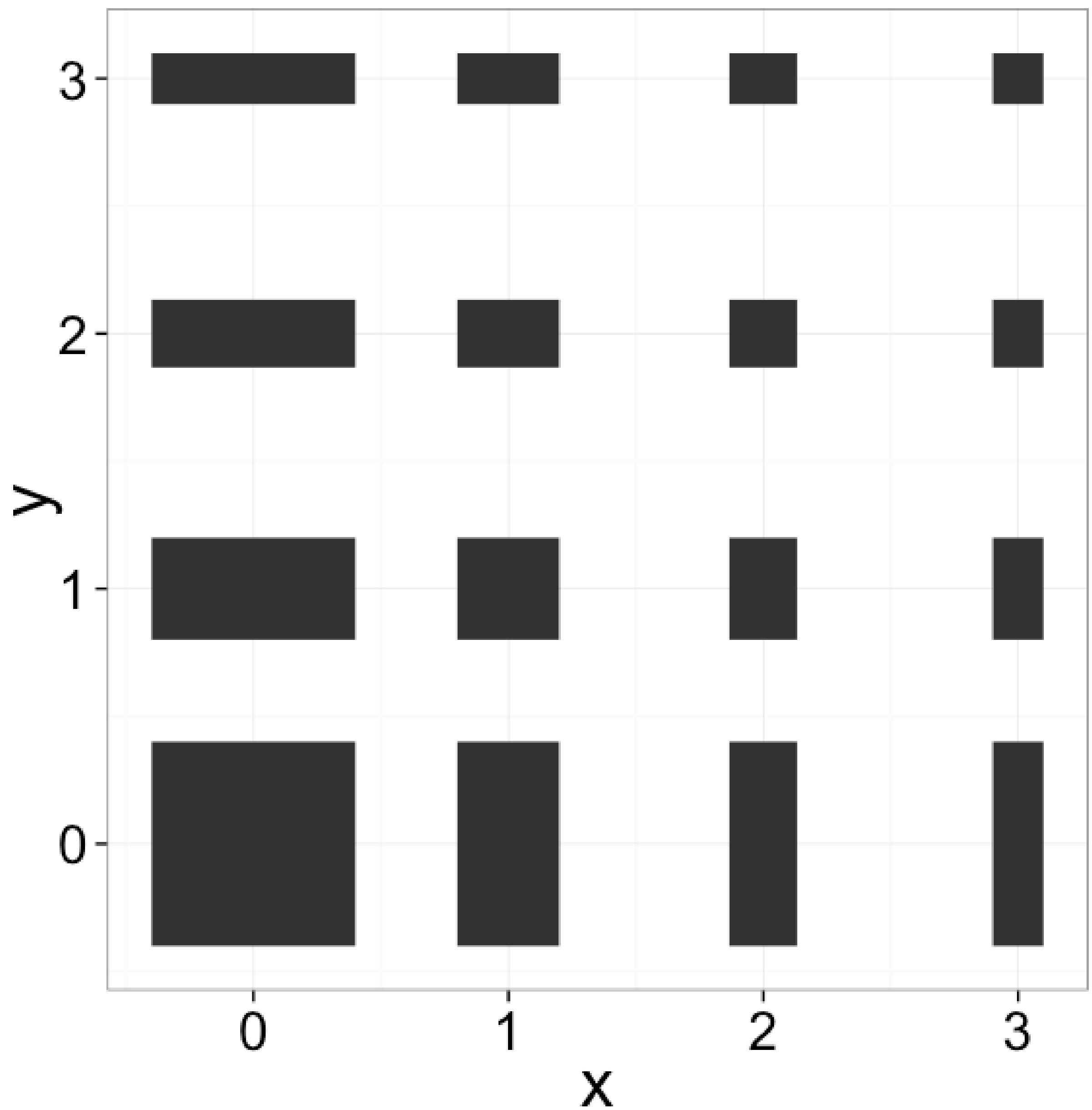
While this defines the 1D case well, it is difficult to apply such metrics to voxel, shape, and tensor data.

Strain Tensor

There are a number of different ways to calculate strain and the strain tensor, but the most applicable for general image based applications is called the infinitesimal strain tensor (http://en.wikipedia.org/wiki/Infinitesimal_strain_theory), because the element matches well to square pixels and cubic voxels.



A given strain can then be applied and we can quantify the effects by examining the change in the small element.



Types of Strain

We catagorize the types of strain into two main catagories:

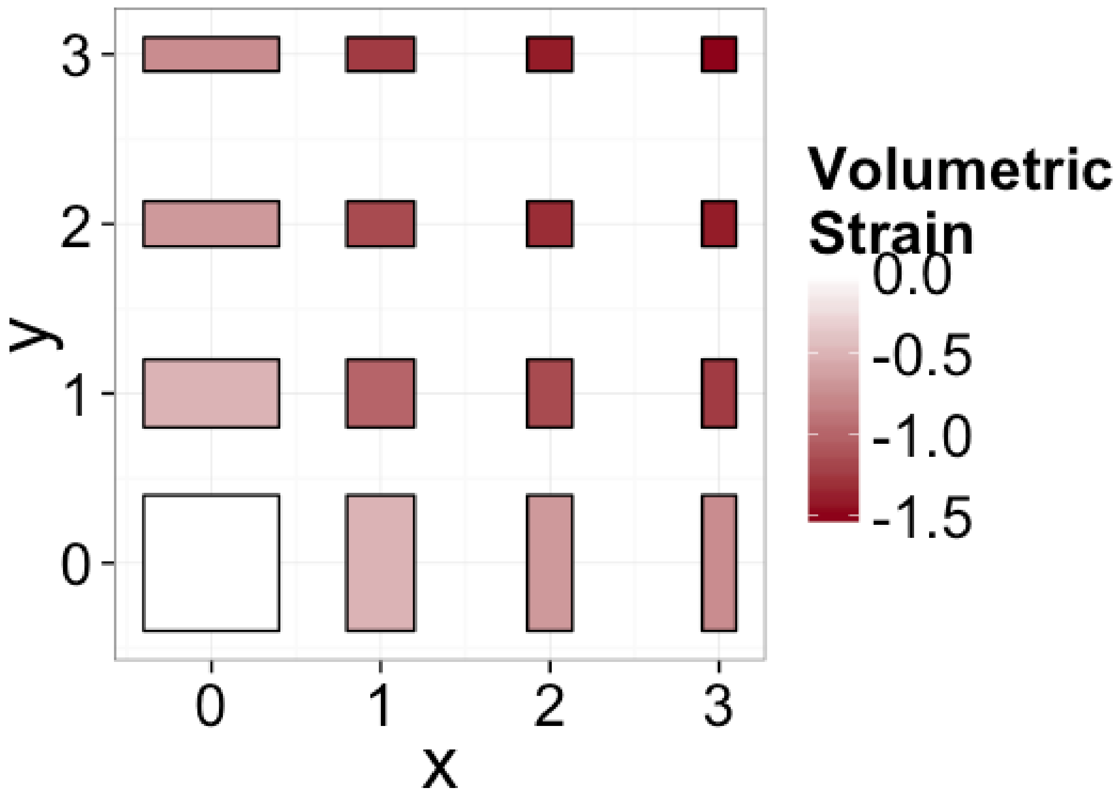
$$\underbrace{\mathbf{E}}_{\text{Total Strain}} = \underbrace{\varepsilon_M \mathbf{I_3}}_{\text{Volumetric}} + \underbrace{\mathbf{E}'}_{\text{Deviatoric}}$$

Volumetric / Dilational

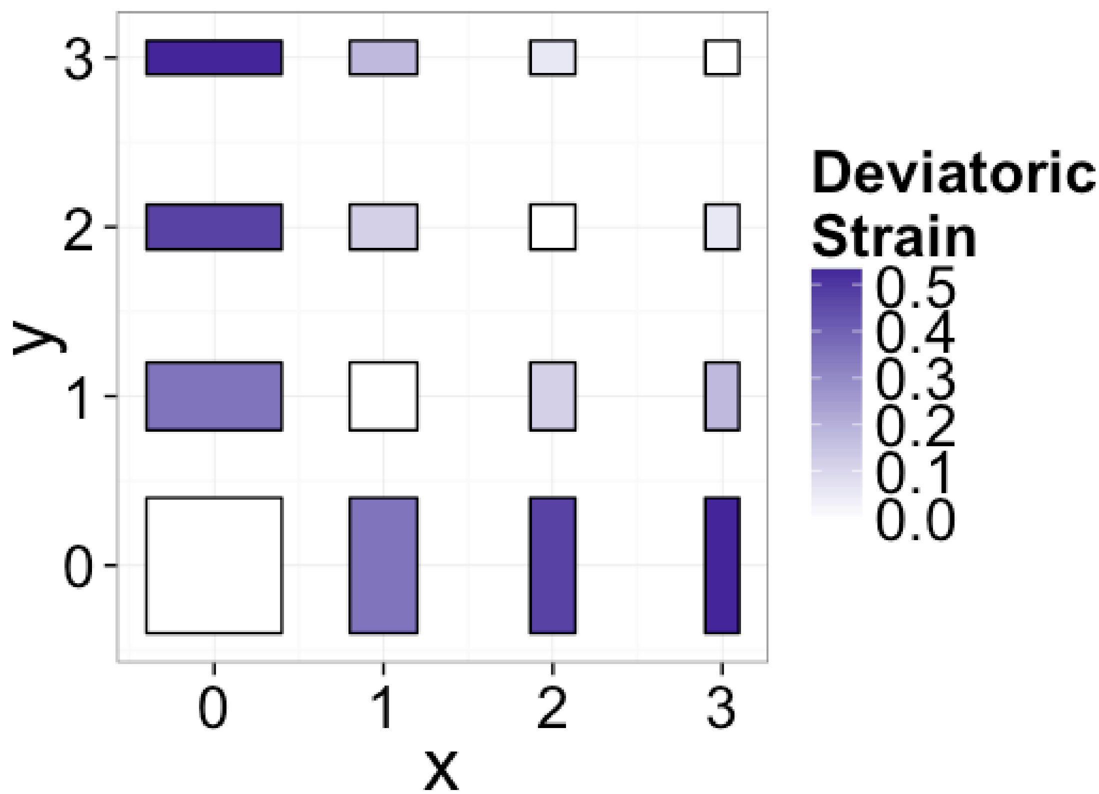
The isotropic change in size or scale of the object.

Deviatoric

The change in the proportions of the object (similar to anisotropy) independent of the final scale



+/- R Code



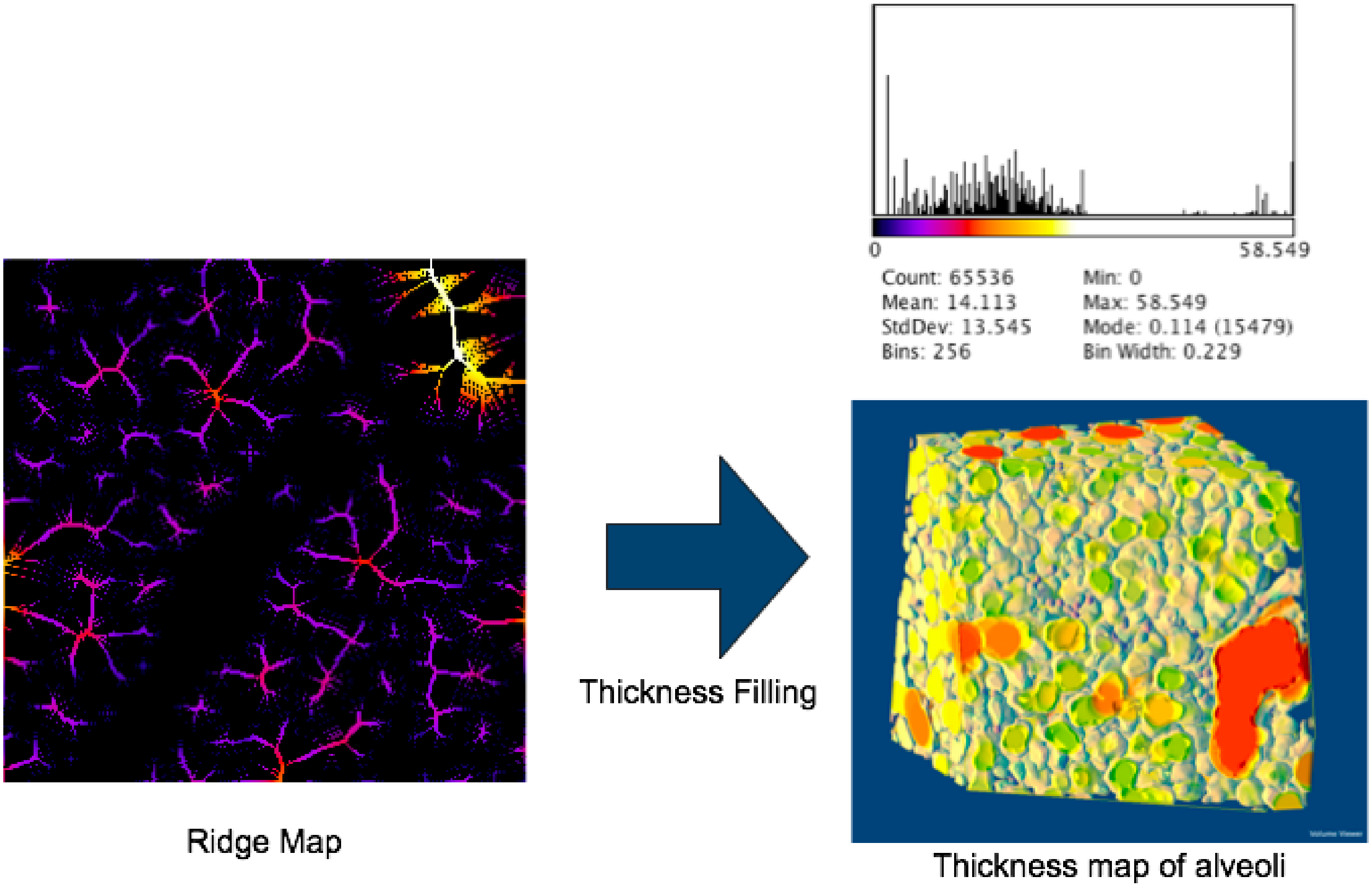
Strain Tensor

This can then be expressed more concretely in the form of a strain tensor which can be broken down into volumetric and deviatoric strain

$$\mathbf{E} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix}$$

Thickness - Lung Tissue

__ Data provided by Goran Lovric and Rajmund Mokso __ Since lung



Curvature - Metal Systems

__ Data provided by Julie Fife __

Two Point Correlation - Volcanic Rock

__ Data provided by Mattia Pistone and Julie Fife __ The air phase changes from small very anisotropic bubbles to one large connected pore network. The same tools cannot be used to quantify those systems. Furthermore there are motion artifacts which are difficult to correct.

We can utilize the two point correlation function of the material to characterize the shape generically for each time step and then compare.