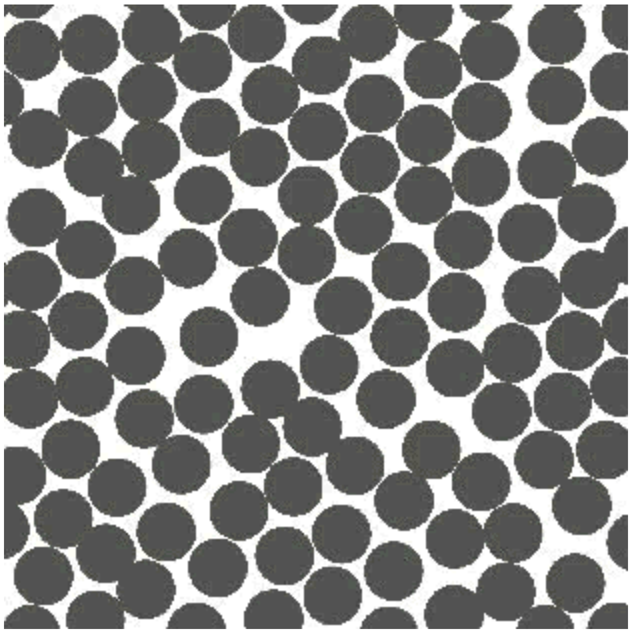
Image Analysis



Mark D. Shattuck
Benjamin Levich Institute
City College of New York
NSF-DMR, NSF-CBET, NSF-CMMI

Hands On 2015

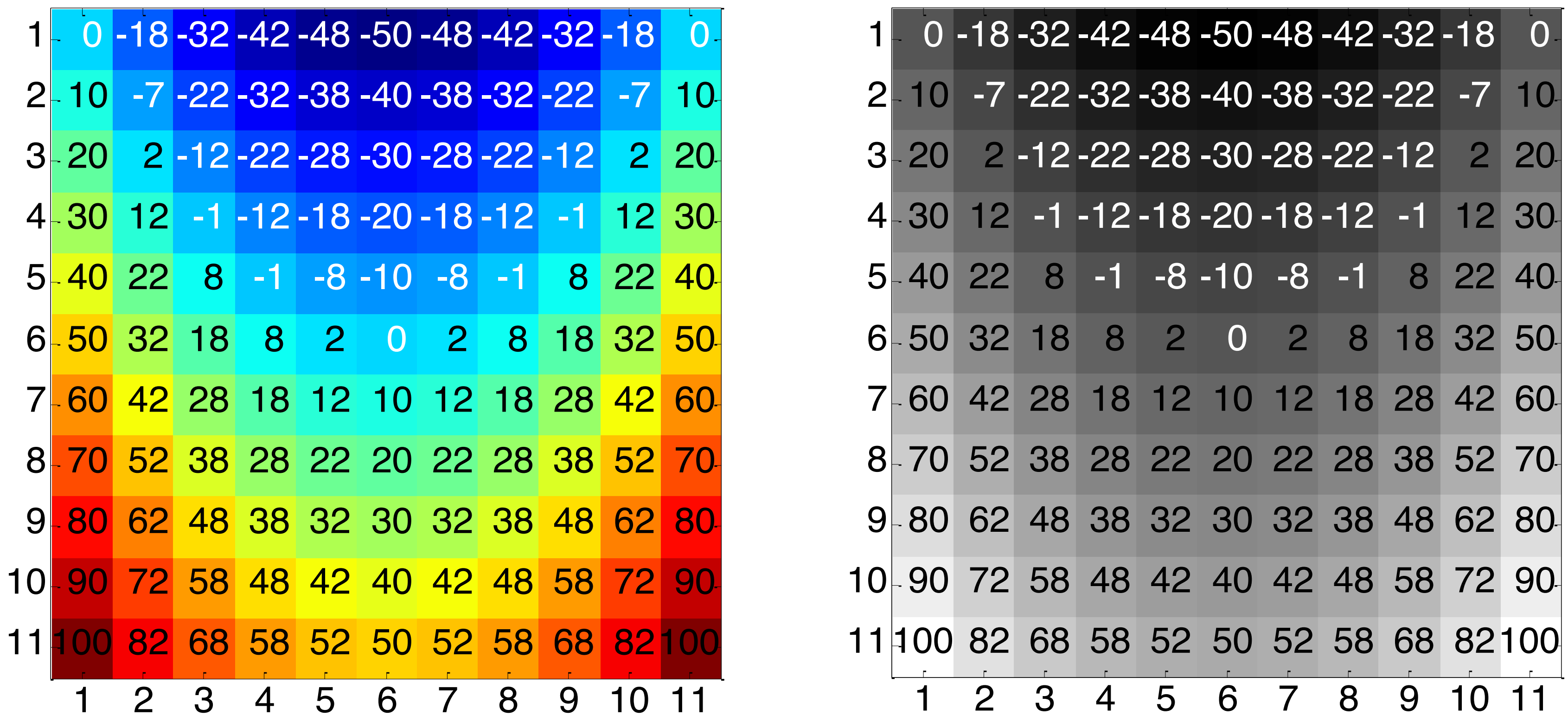
Introduction

- The brain is the best image analyzer.
- If you *cannot* do it with your brain, you *cannot* do with a computer.
- If you *can* do it with your brain, you can rarely do it with a computer.
- Making you image better will pay off.

Outline

- Image Analysis tools
  - Fast Fourier Transform (FFT)
  - Convolution and Cross-correlation
- Particle Imaging Velocimetry (PIV)
- Particle Tracking
- Oscillatory Image Demodulation (OID)

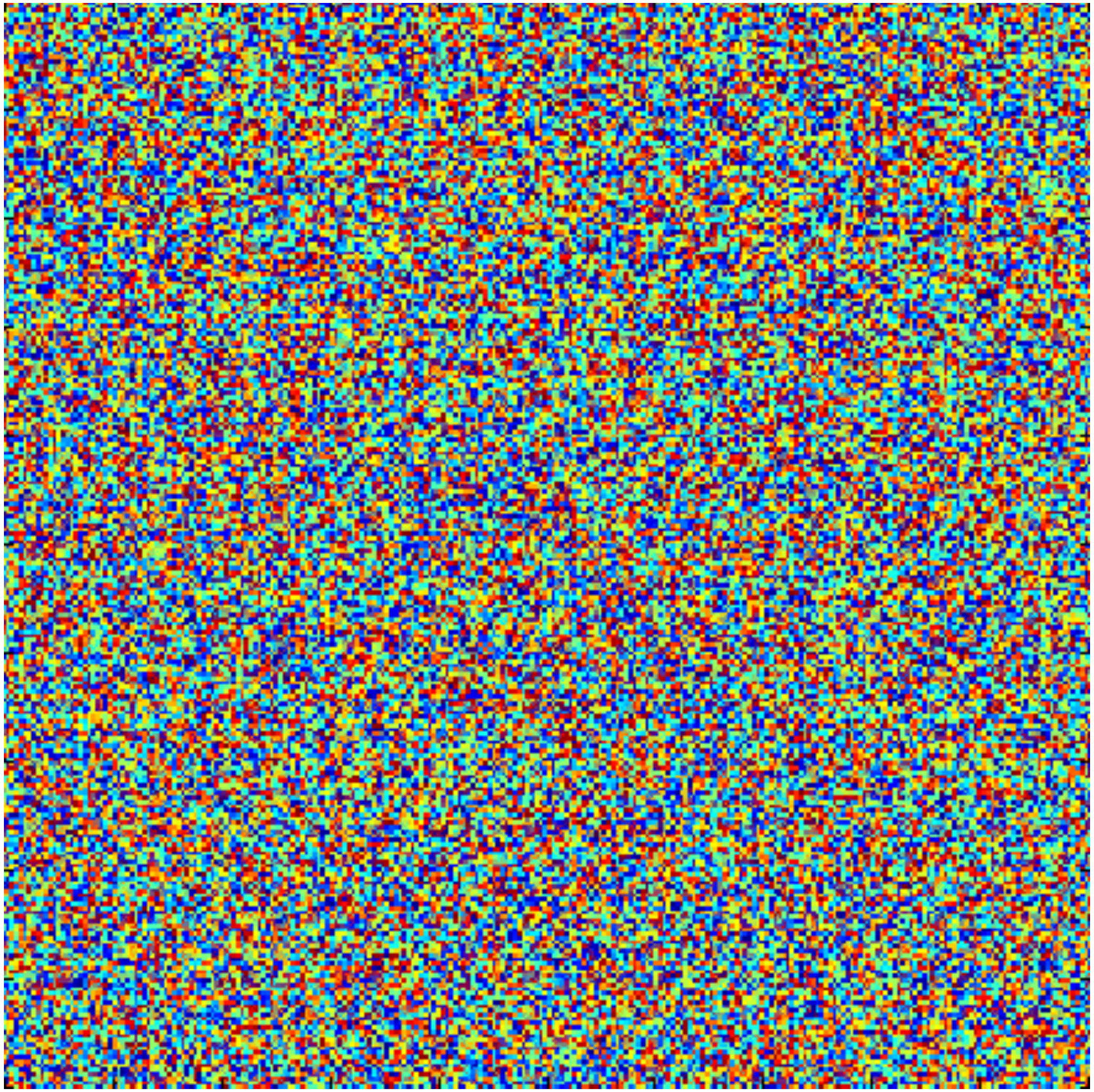
Image



0 -18 -32 -42 -48 -50 -48 -42 -32 -18 0
10 -7 -22 -32 -38 -40 -38 -32 -22 -7 10
20 2 -12 -22 -28 -30 -28 -22 -12 2 20
30 12 -1 -12 -18 -20 -18 -12 -1 12 30
40 22 8 -1 -8 -10 -8 -1 8 22 40
50 32 18 8 2 0 2 8 18 32 50
60 42 28 18 12 10 12 18 28 42 60
70 52 38 28 22 20 22 28 38 52 70
80 62 48 38 32 30 32 38 48 62 80
90 72 58 48 42 40 42 48 58 72 90
100 82 68 58 52 50 52 58 68 82 100
1 2 3 4 5 6 7 8 9 10 11

$$I_{nm}$$

Video Image Data



$$I_{nm}(k\Delta t)$$

FFT

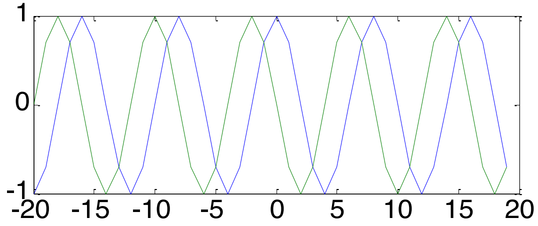
- Fast Fourier Transform

$$F_{nm} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} f_{kl} e^{-2\pi i(kn/K + lm/L)}$$

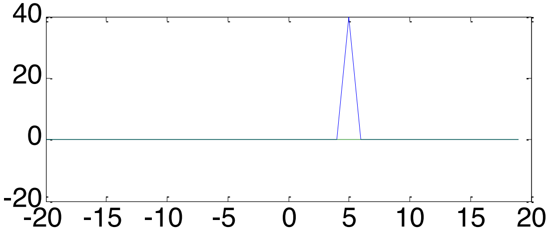
$$f_{kl} = \frac{1}{NM} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} F_{nm} e^{+2\pi i(kn/N + lm/M)}$$

FFT

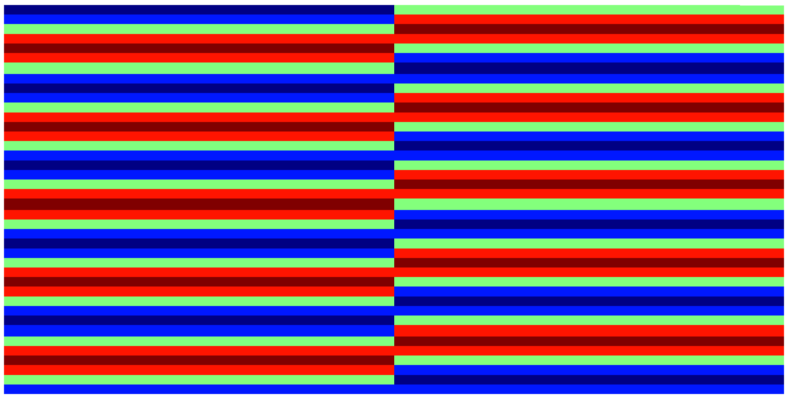
$$f_k = e^{2\pi i k n_0 / N}$$



$$F_n = N\delta_{nn_0}$$

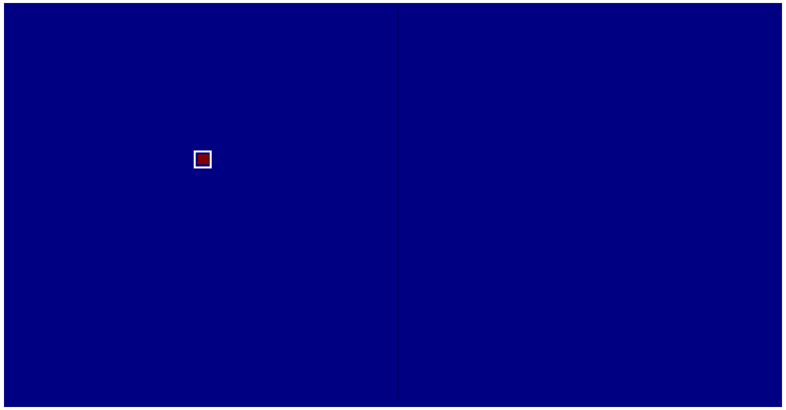


$$f_{kl} = e^{2\pi i k n_0 / N}$$



Re Im

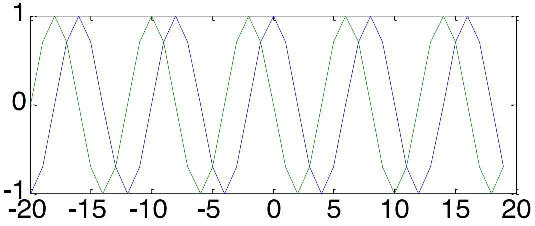
$$F_{nm} = NM\delta_{nn_0}\delta_{m0}$$



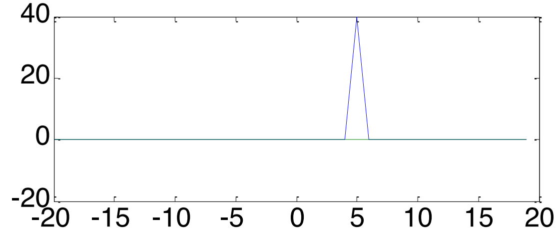
Re Im

FFT

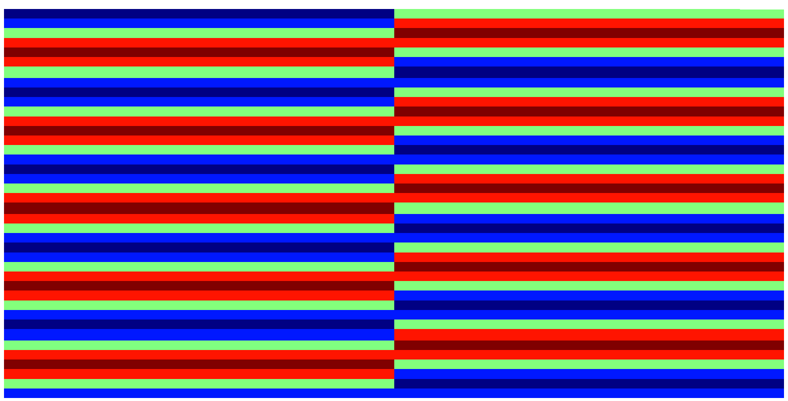
$$f_k = e^{2\pi i k n_0 / N}$$



$$F_n = N\delta_{nn_0}$$

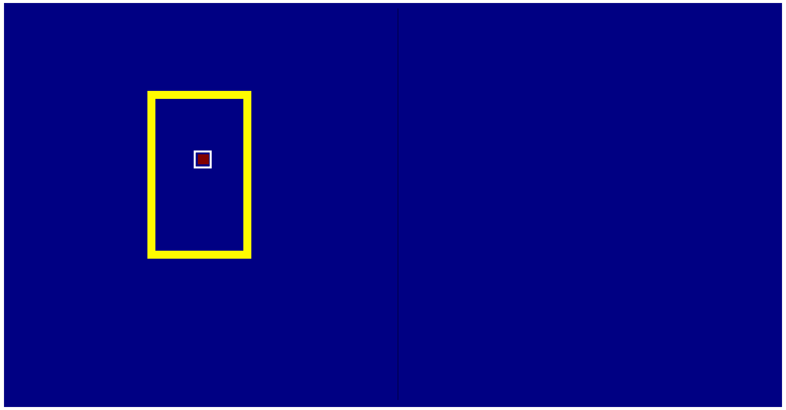


$$f_{kl} = e^{2\pi i k n_0 / N}$$



Re Im

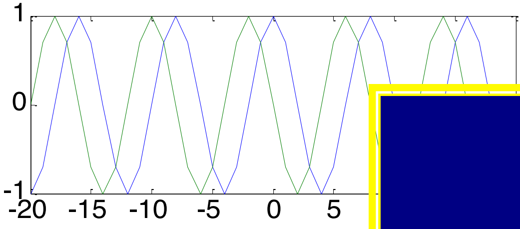
$$F_{nm} = NM\delta_{nn_0}\delta_{m0}$$



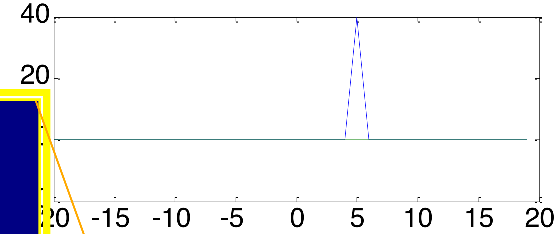
Re Im

FFT

$$f_k = e^{2\pi i k n_0 / N}$$

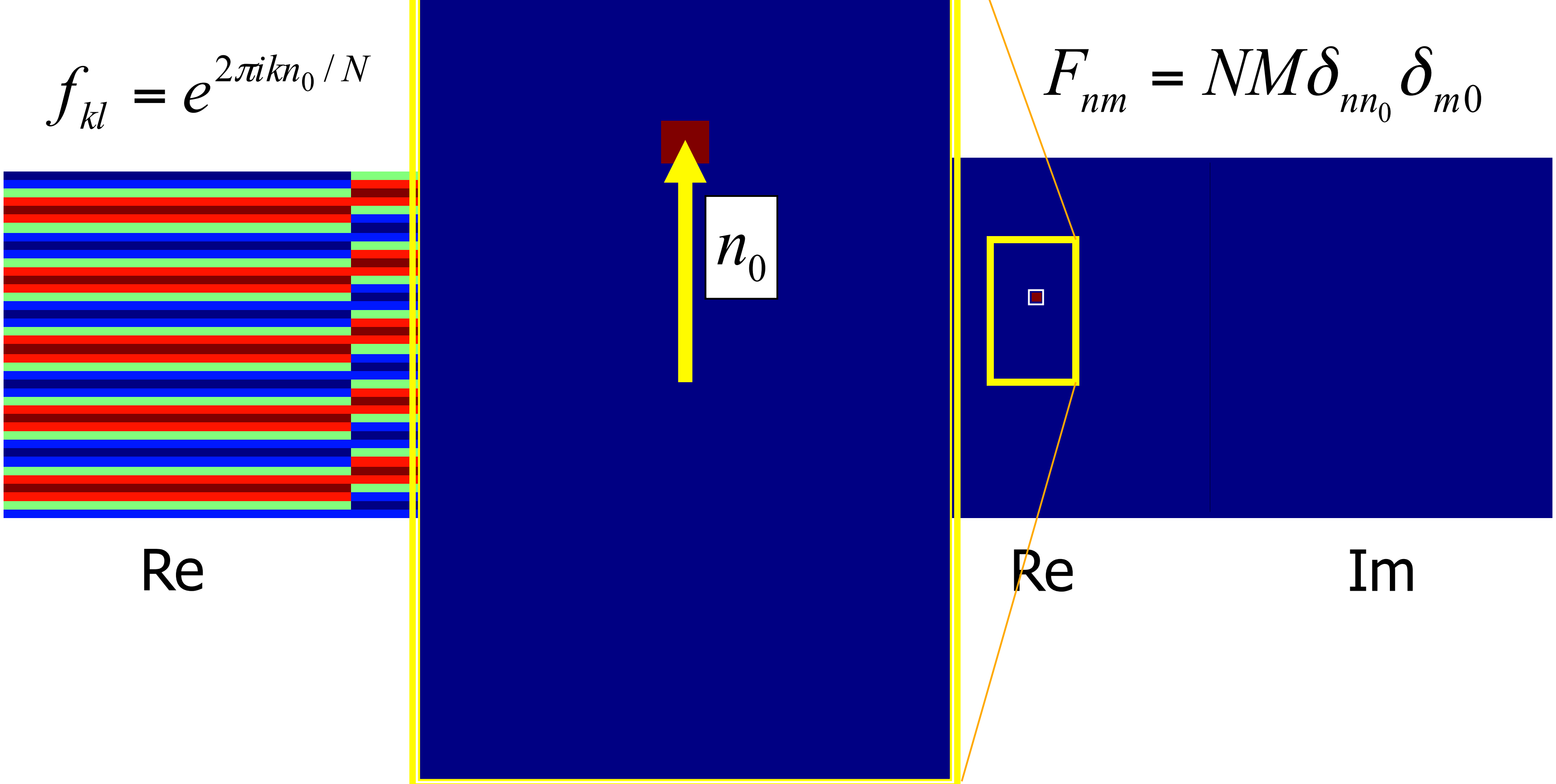


$$F_n = N\delta_{nn_0}$$

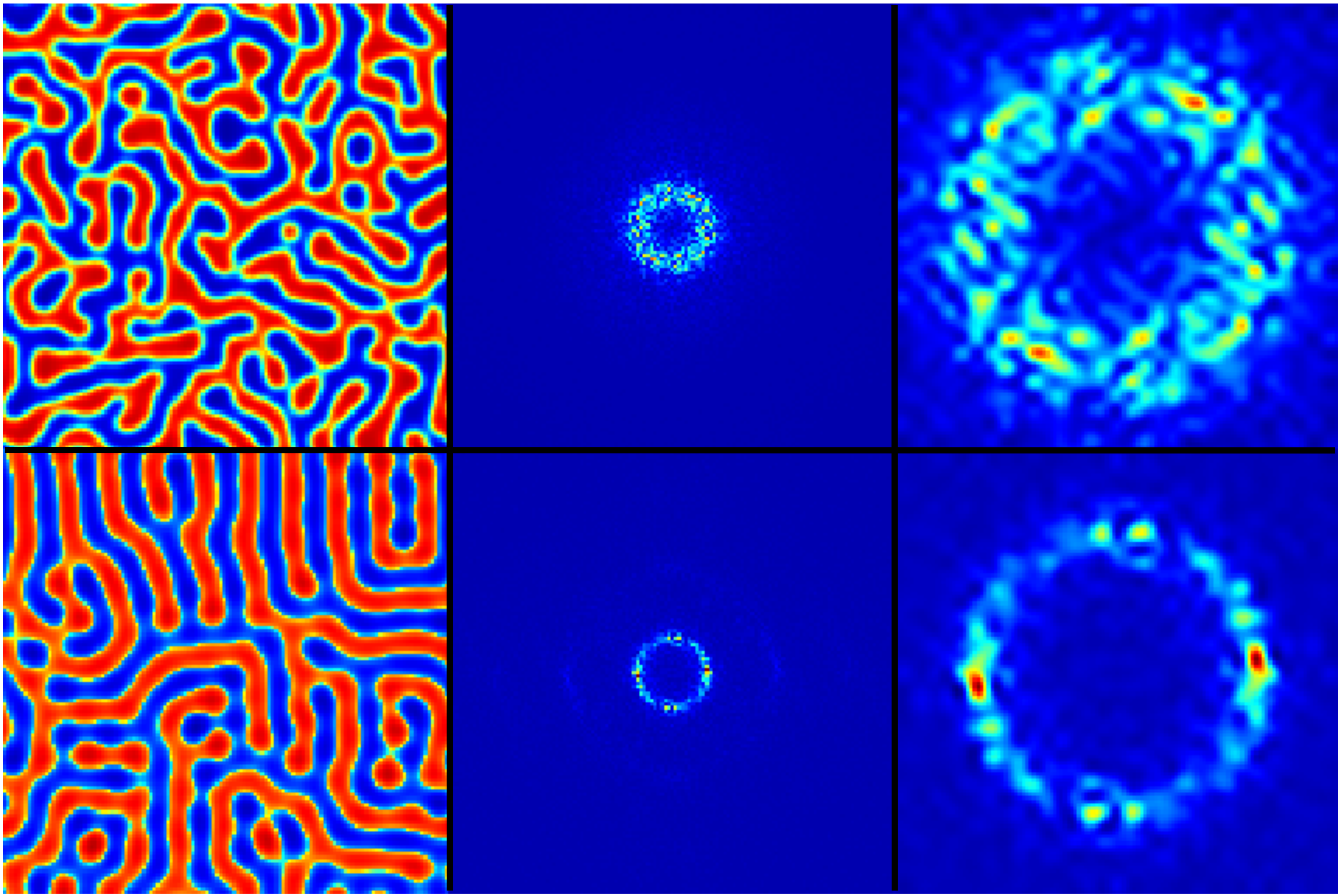


$$f_{kl} = e^{2\pi i k n_0 / N}$$

$$F_{nm} = NM\delta_{nn_0}\delta_{m0}$$



FFT



Convolution

- Convolution and Cross Correlation

$$C(f,g)_n = \sum_{k=0}^{K-1} f_k g_{(n-k)}$$

$$X(f,g)_n = \sum_{k=0}^{K-1} f_k^* g_{(n+k)}$$

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3 4

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3 4 7

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3 4 7 7

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3 4 7 7 2

Convolution

1 3 2 1 3 1 0 2

1 3 2 1 3

2 0 1

1 3 4 7 7 2 6

Convolution

1 3 2 1 3 　　　　　　　　1 0 2

1 3 2 1 3

2 0 1

1 3 4 7 7 2 6

full

same

valid

Convolution

- Convolution and Cross Correlation

$$C(f,g)_n = \sum_{k=0}^{K-1} f_k g_{(n-k)}$$

$$X(f,g)_n = \sum_{k=0}^{K-1} f_k^* g_{(n+k)}$$

Correlation

1 3 2 1 3   1 0 2

1 3 2 1 3

1 0 2

2

Correlation

1 3 2 1 3   1 0 2

1 3 2 1 3

1 0 2

2 6

Correlation

1 3 2 1 3 1 0 2

1 3 2 1 3

1 0 2

2 6 5

Correlation

1 3 2 1 3

1 0 2

1 3 2 1 3

1 0 2

2 6 5 5

Correlation

1 3 2 1 3   1 0 2

1 3 2 1 3

1 0 2

2 6 5 5 8

Correlation

1 3 2 1 3 1 0 2

1 3 2 1 3

1 0 2

2 6 5 5 8 1

Correlation

1 3 2 1 3 1 0 2

1 3 2 1 3

1 0 2

2 6 5 5 8 1 3

Correlation

1 3 2 1 3   1 0 2

1 3 2 1 3

1 0 2

2 6 5 5 8 1 3

full

same

valid

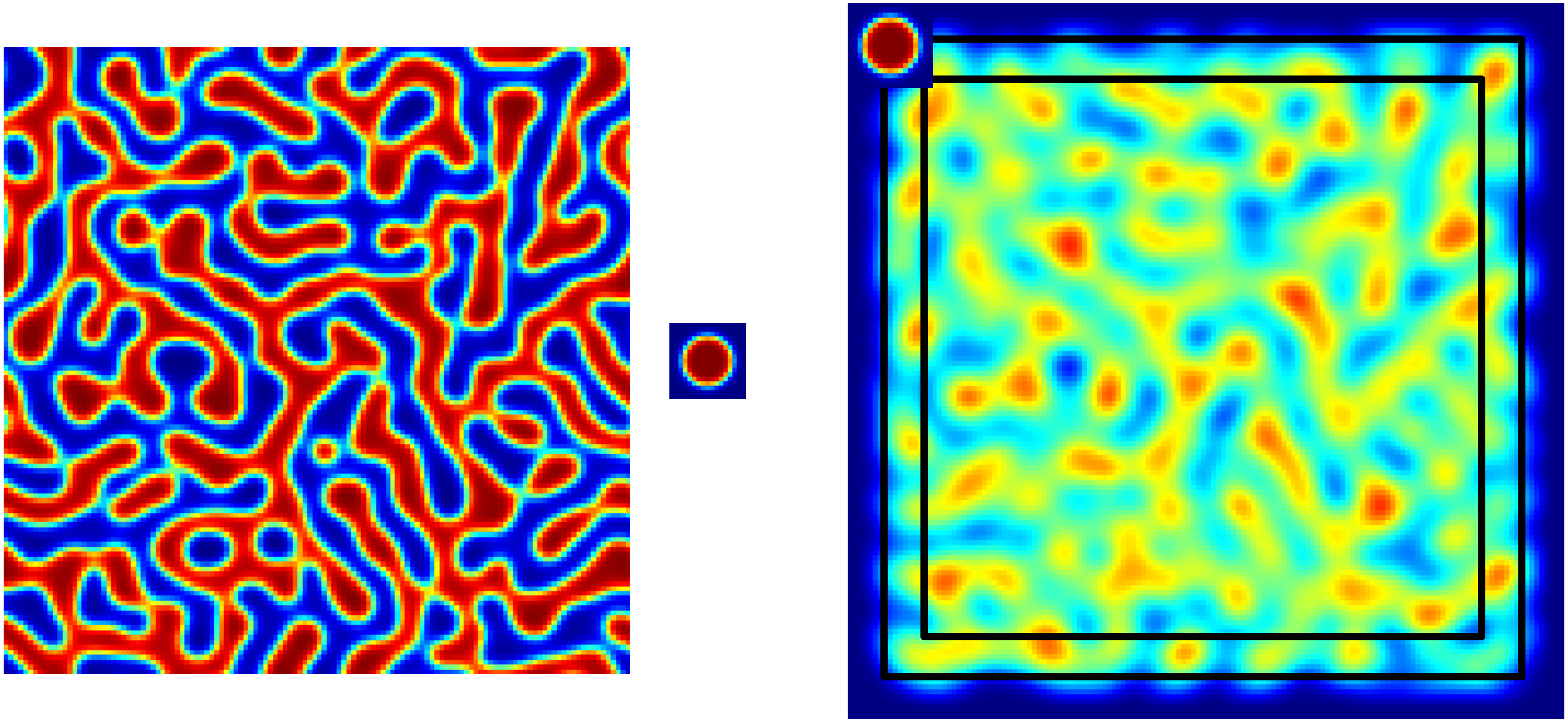
Convolution

- Convolution and Cross Correlation

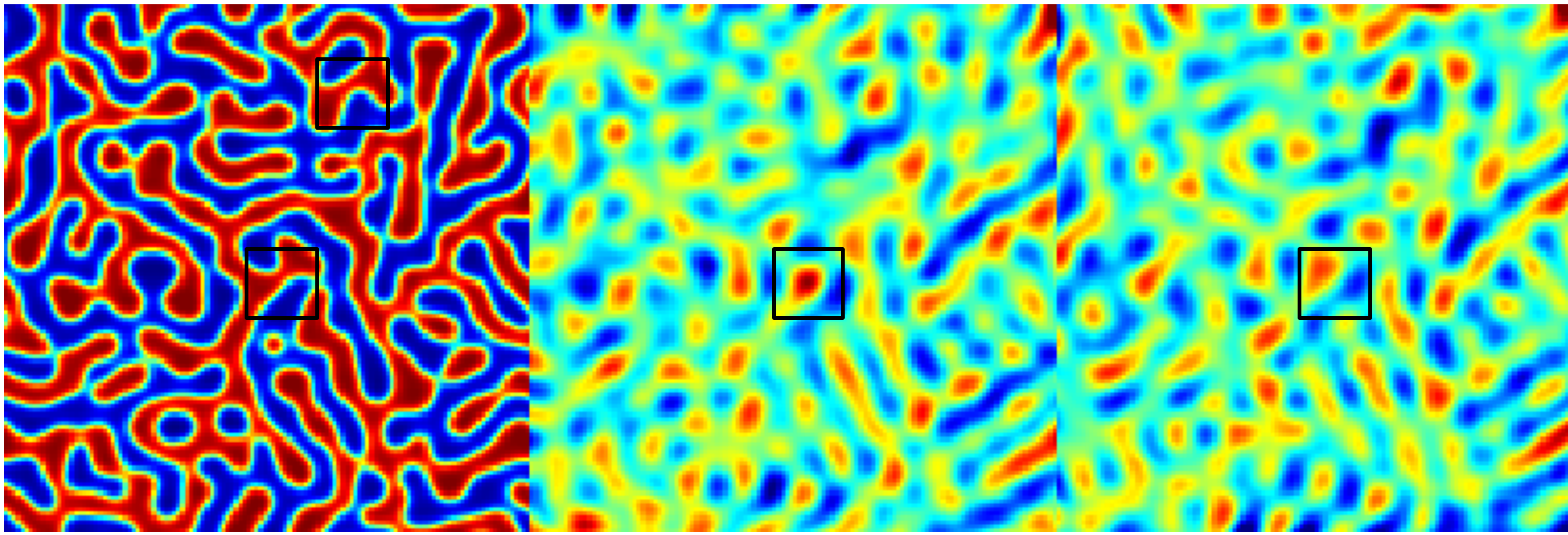
$$C(f,g)_{nm} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} f_{kl} g_{(n-k)(m-l)}$$

$$X(f,g)_{nm} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} f_{kl}^{*} g_{(n+k)(m+l)}$$

Convolution



Correlation vs. Convolution



Correlation Convolution

Least-Square Fit

- Minimum Squared Difference

$$\chi^2(I,p)_{nm} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} [I_{kl} - p_{(n+k)(m+l)}]^2$$

$$\chi^2(I,p)_{nm} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} [I_{kl}]^2 - 2 I_{kl} p_{(k+n)(l+m)} + [p_{(k+n)(l+m)}]^2$$

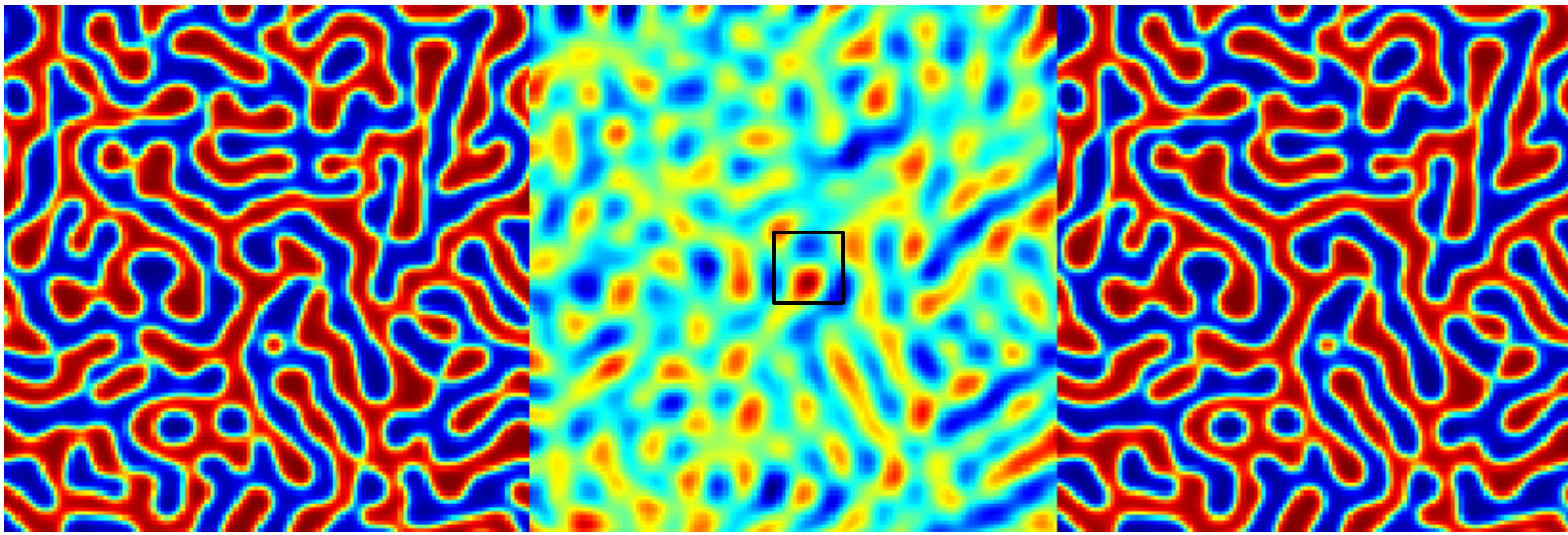
Correlation

$$\min\big|_{nm} \left\{ \chi^2(I,p)_{nm} \right\}$$

Particle Imaging Velocimetry

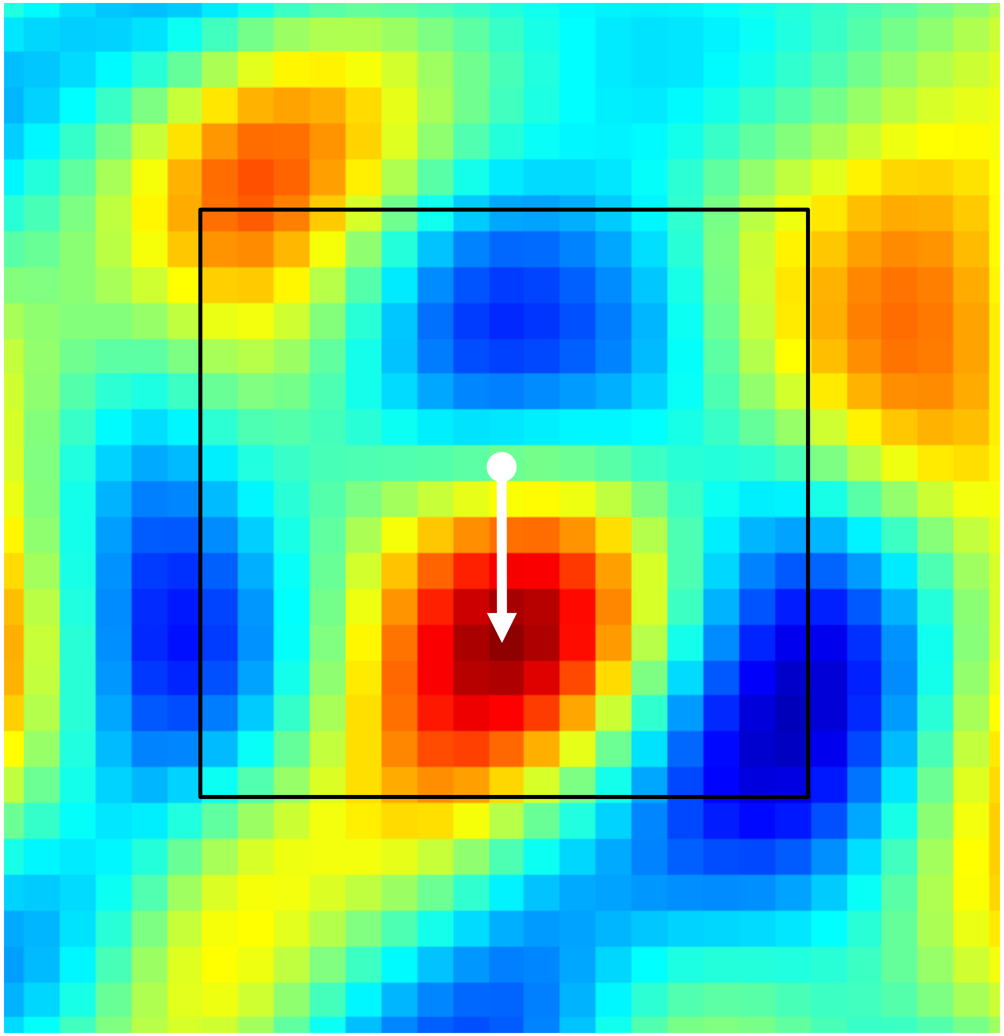
- Take a small patch, p from image n and cross-correlate it with image n+k.
- Find position of maximum.
- Distance from origin of p to maximum is the best fit for the displacement in time k.
- Pick new patch and repeat until all patches from frame n are found.
- Go to frame n+1 and repeat until all frames are done.

PIV

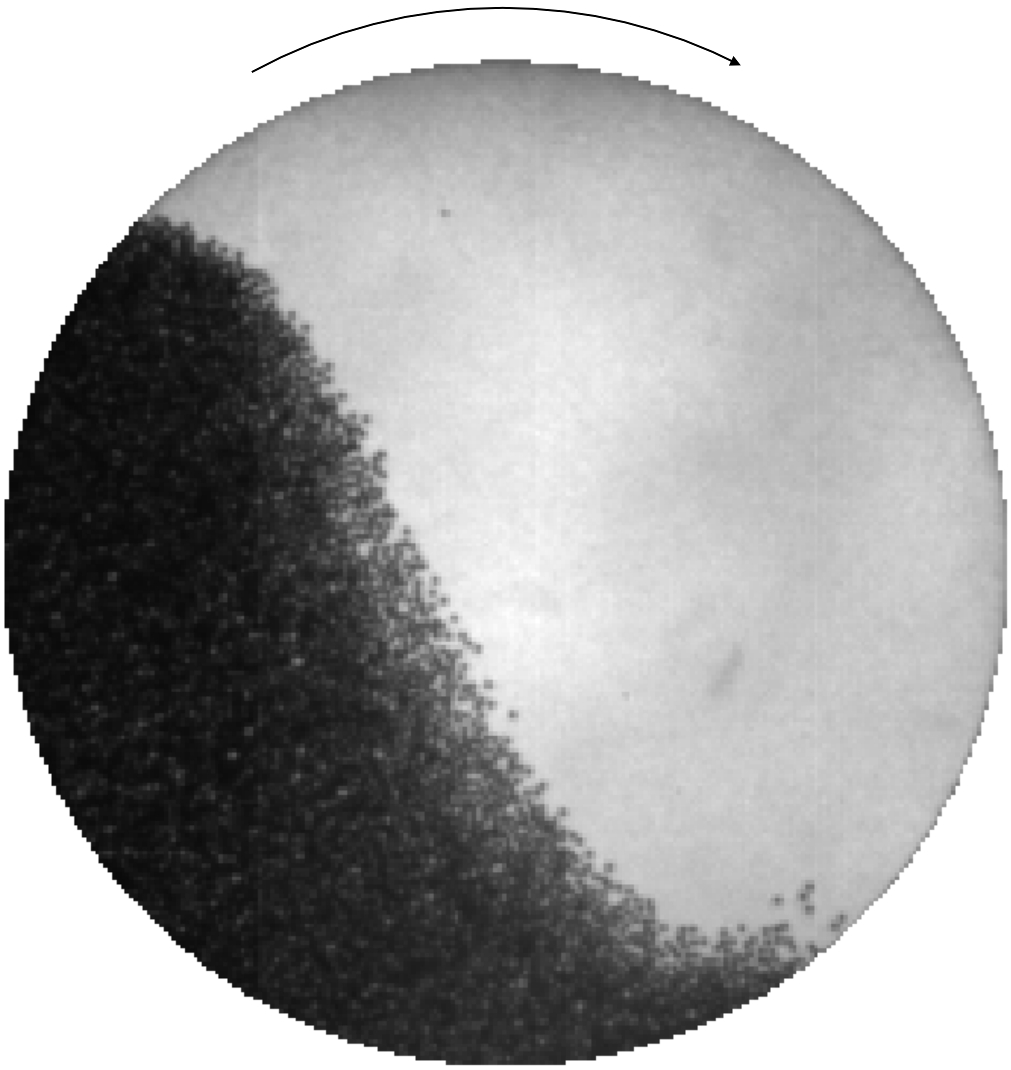


Correlation

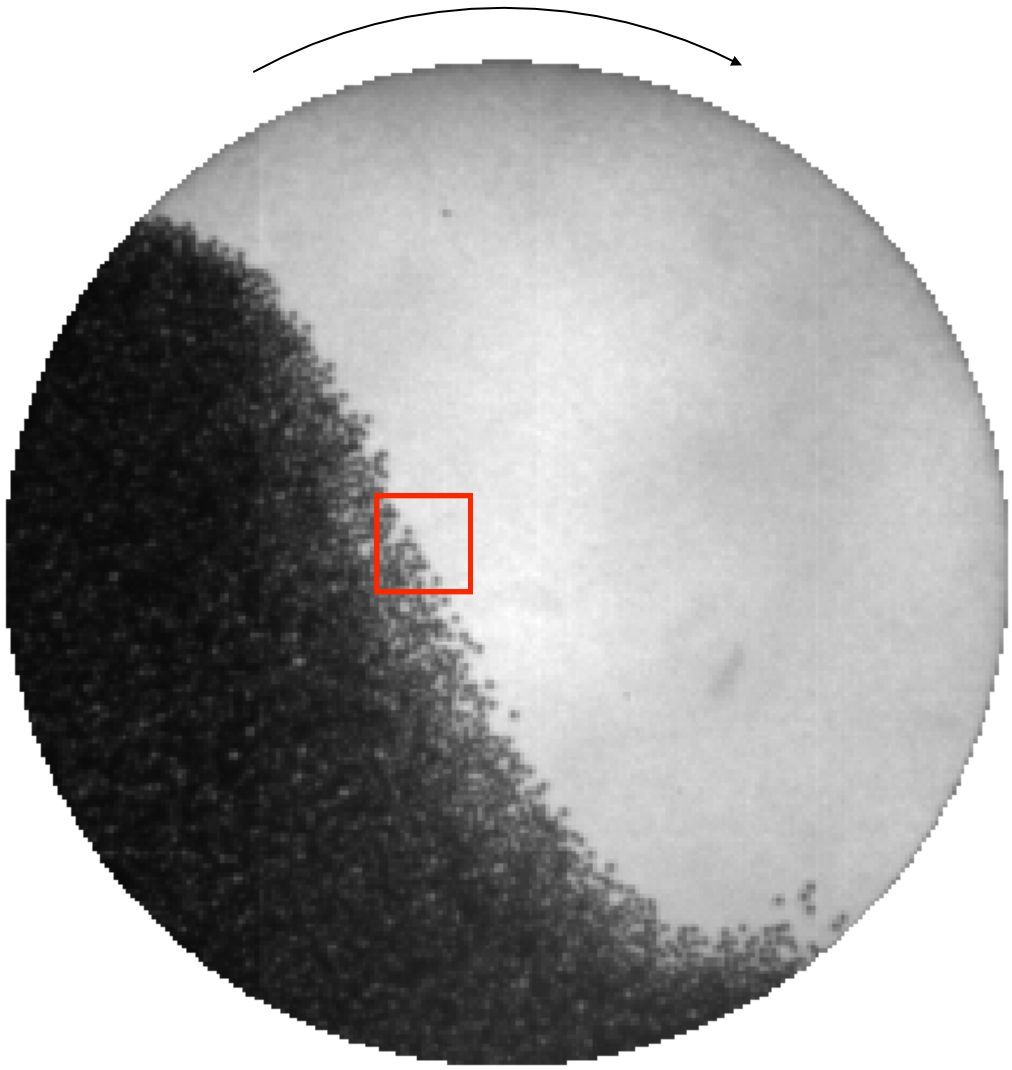
PIV



2D Rotating Drum

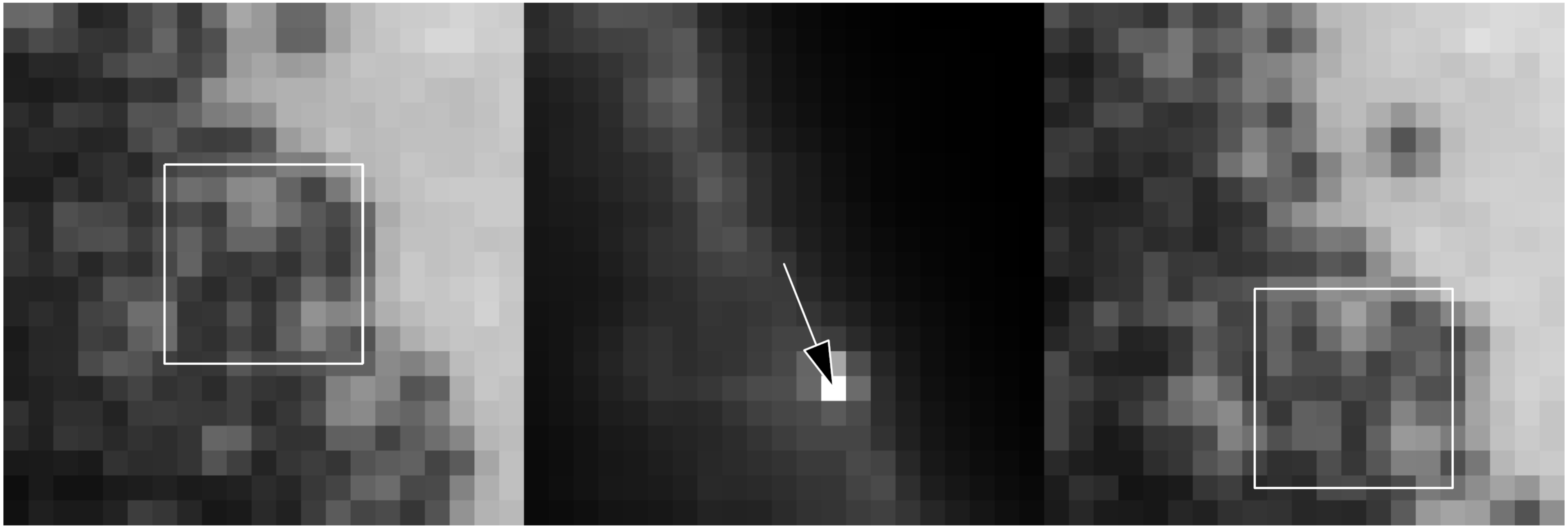


2D Rotating Drum



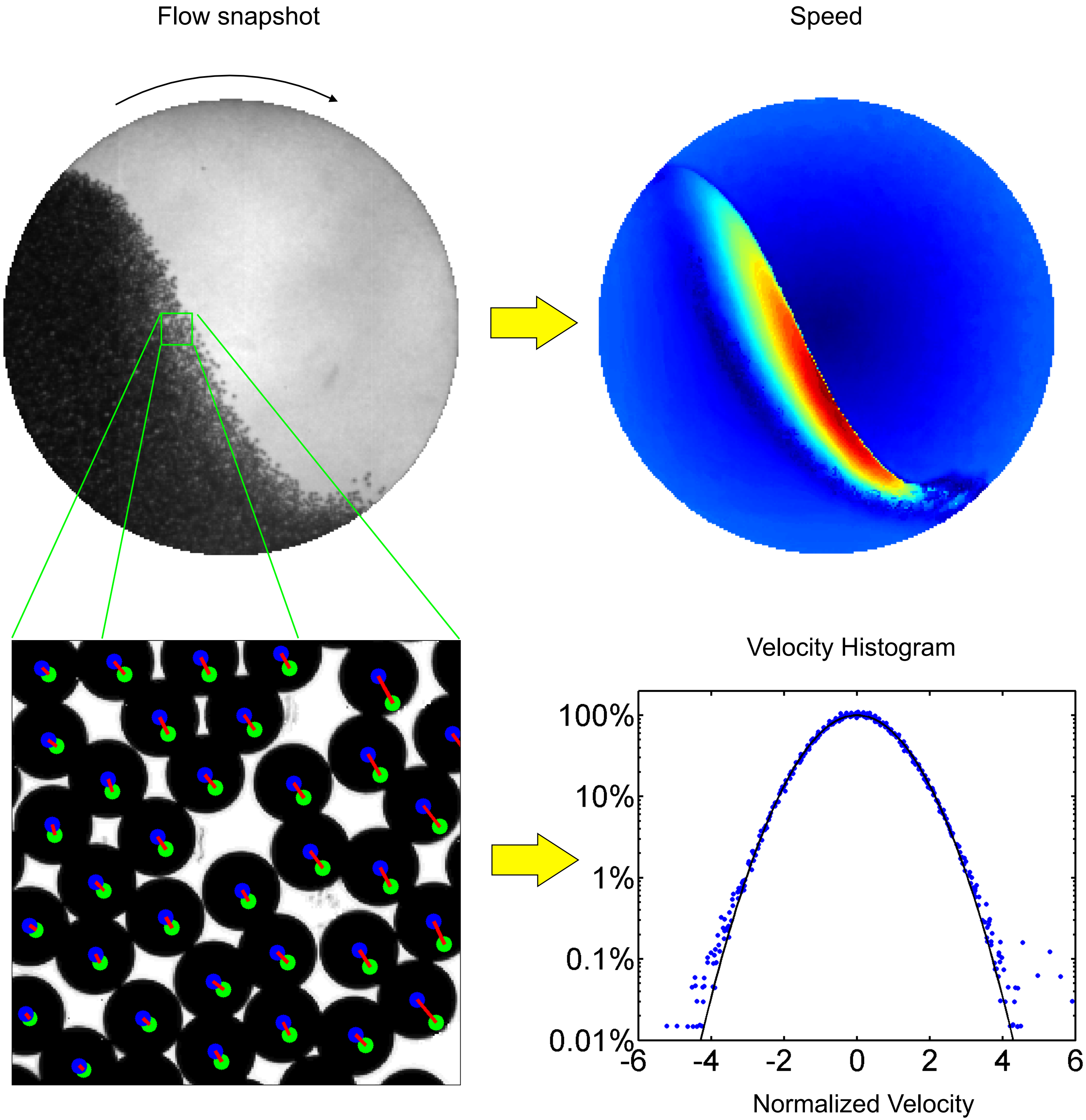
PIV

X-Corr

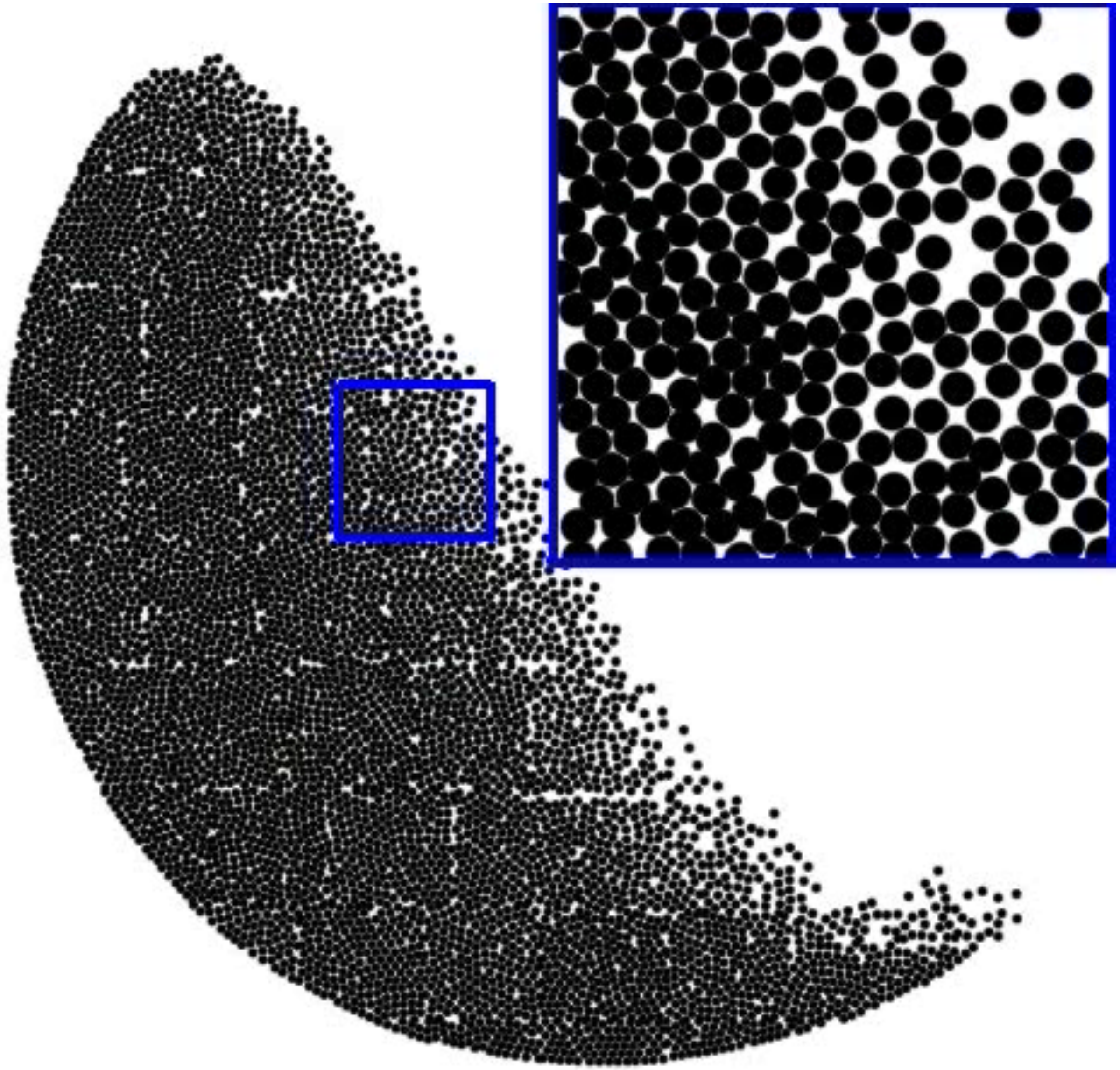


Frame n

Frame n+1


Flow snapshot
Speed
Velocity Histogram
100%
10%
1%
0.1%
0.01%
-6
-4
-2
0
2
4
6
Normalized Velocity

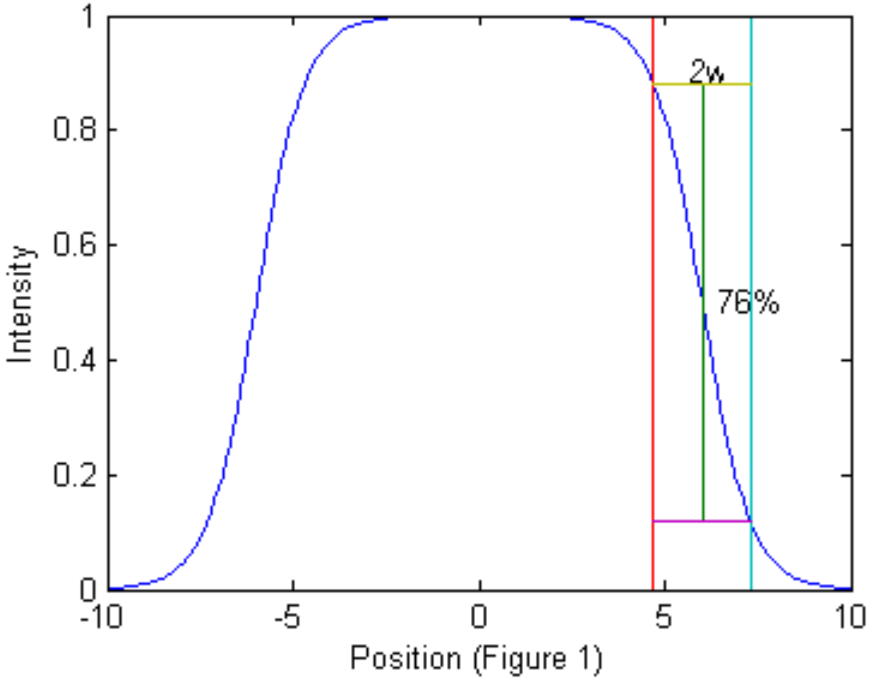
2D Rotating Drum

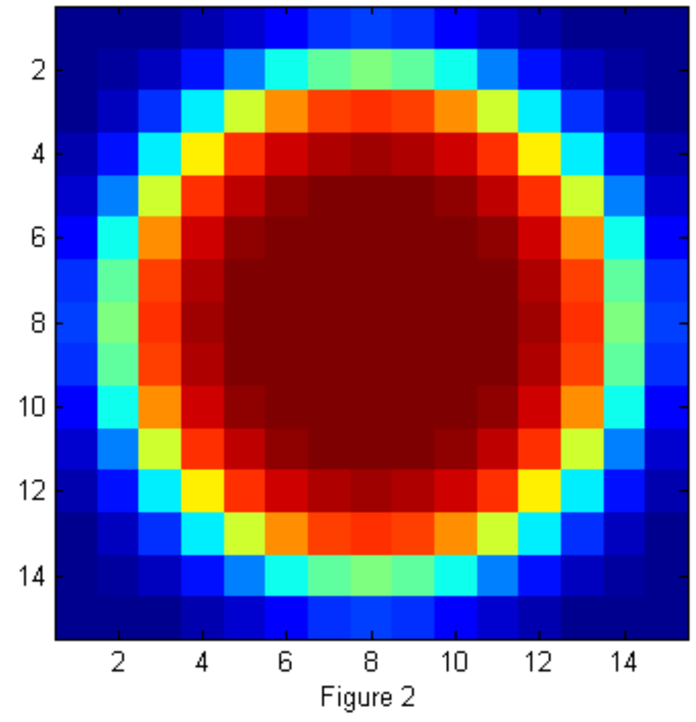


Particle Tracking
Pixel Accurate

$$I_c(\vec{x}) = \sum_{n=1}^{N} I_p(\vec{x} - \vec{x}_n(t); D, ...),$$

$$I_p(\vec{x}; D, w) = [1 - tanh(\frac{|\vec{x}| - D/2}{w})]/2.$$





Particle Tracking

Pixel Accurate

$$\chi^2(\vec{x}_0; D, w) = \int W(\vec{x} - \vec{x}_0)[I(\vec{x}) - I_p(\vec{x} - \vec{x}_0; D, w)]^2 \, d\vec{x},$$

$$\chi^2(\vec{x}_0; D, w) = \int W(\vec{x} - \vec{x}_0)[I(\vec{x})^2 - 2I(\vec{x})I_p(\vec{x} - \vec{x}_0; D, w) + I_p(\vec{x} - \vec{x}_0; D, w)^2] \, d\vec{x},$$

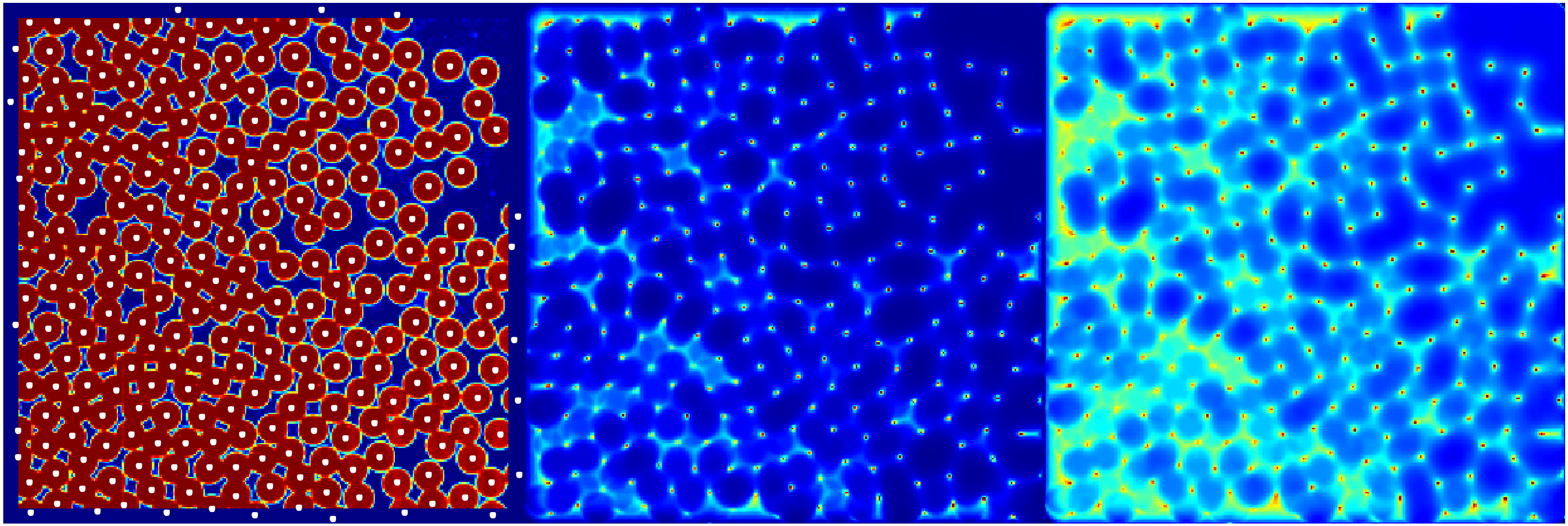
$$\chi^2(\vec{x}_0; D, w) = I^2 \otimes W - 2I \otimes (W I_p) + < W I_p^2 >,$$

$$W = 1; \quad \chi^2(\vec{x}_0; D, w) = \int I^2 \, d\vec{x} - 2I \otimes I_p + < I_p^2 >,$$

$$W = I_p; \quad \chi^2(\vec{x}_0; D, w) = I^2 \otimes I_p - 2I \otimes I_p^2 + < I_p^3 > .$$

Particle Tracking
Pixel Accurate

$$W = I_p; \qquad \chi^2(\vec{x}_0; D, w) = I^2 \otimes I_p - 2I \otimes I_p^2 + < I_p^3 > .$$

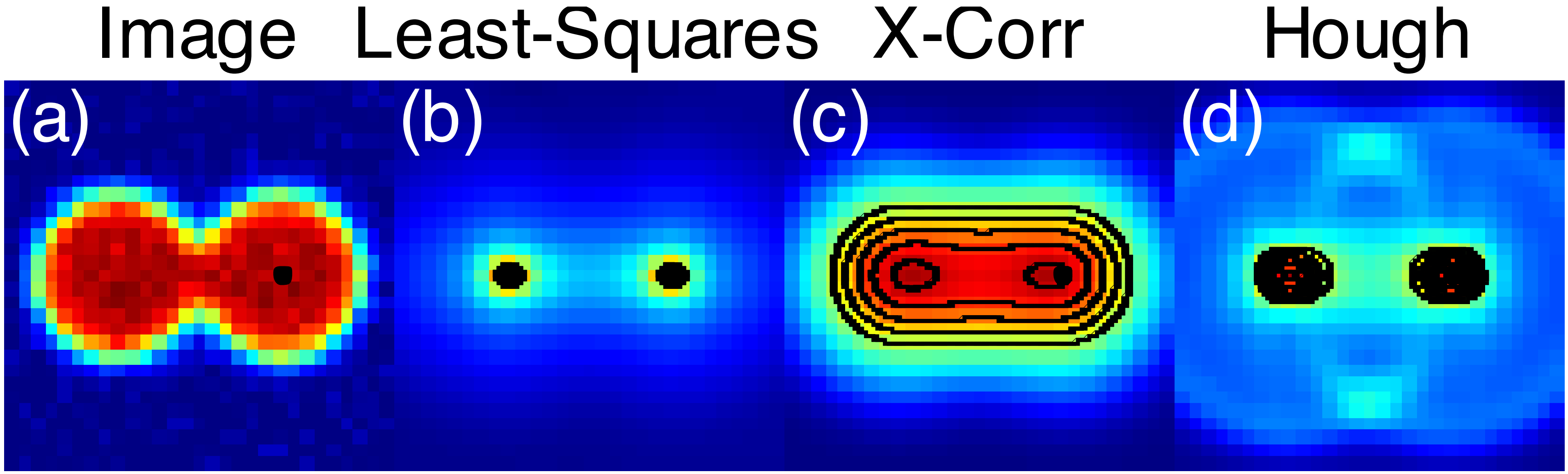


Corrected Image

Chi-Squared Image

Particle Tracking

Pixel Accurate

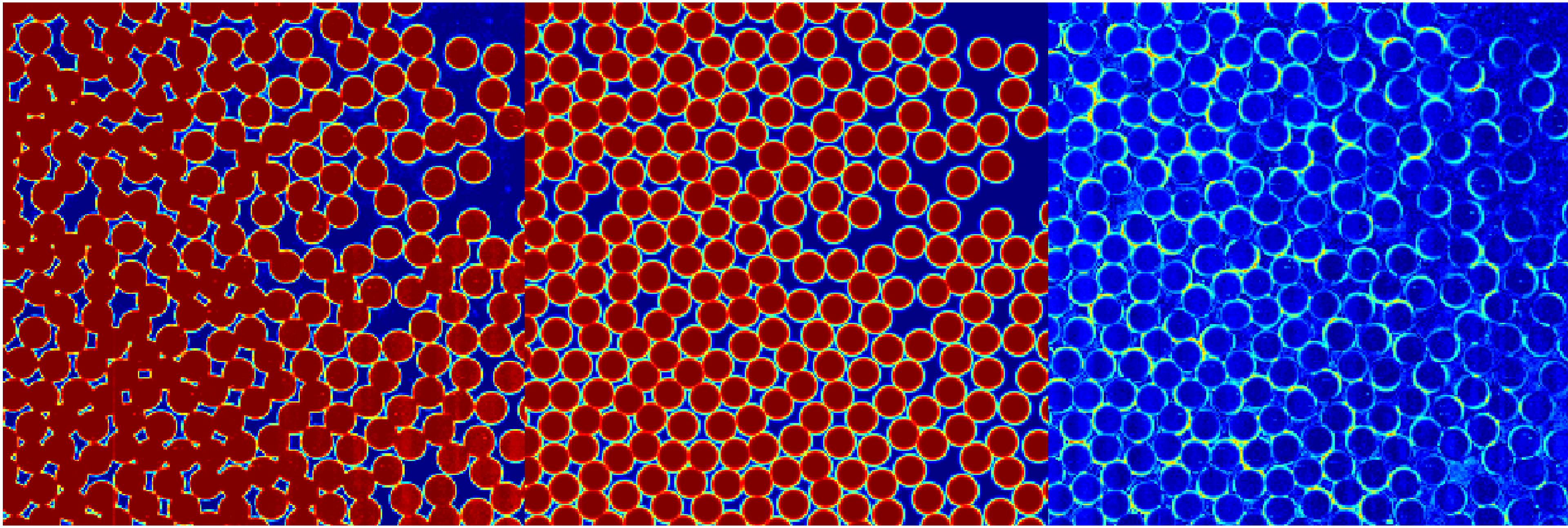


Particle Tracking

Sub-Pixel Accurate

$$\chi^2(\vec{x}_n; D, w) = \int [I(\vec{x}) - I_c(\vec{x}, \vec{x}_n)]^2 \, d\vec{x},$$

$$I_c(\vec{x}, \vec{x}_n) = \sum_n W_n(\vec{x}) I_p(\vec{x} - \vec{x}_n; D, w),$$



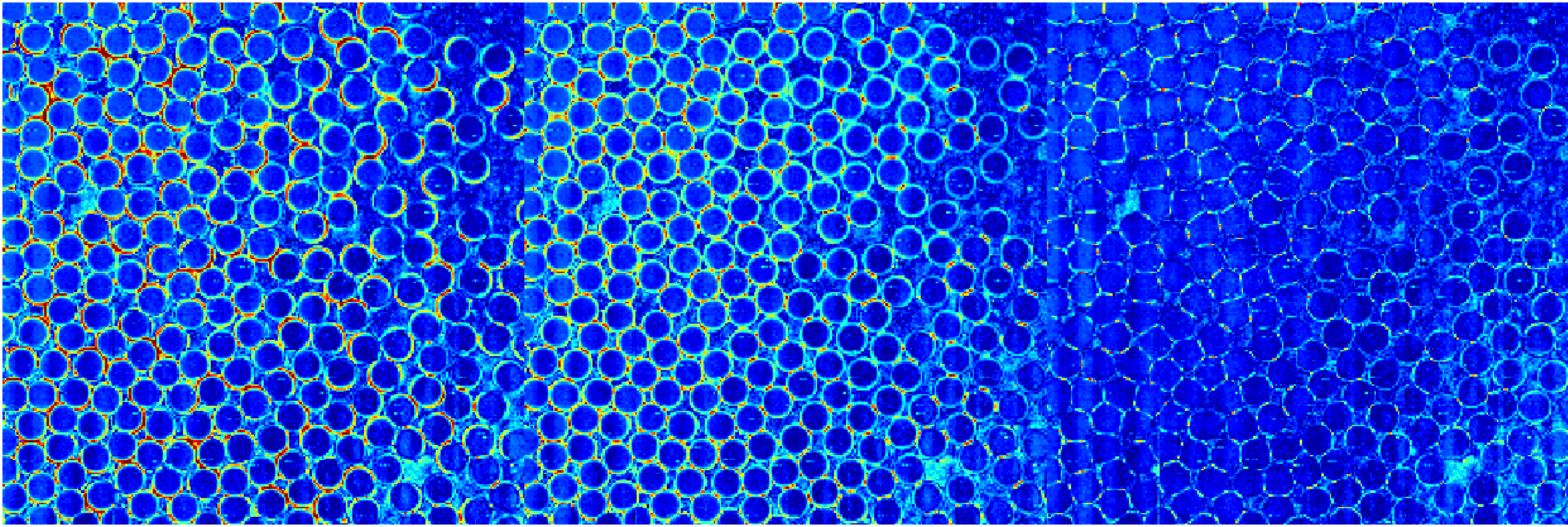
Particle Tracking

Sub-Pixel Accurate

$$\frac{\partial \chi^2(\vec{x}_n^*; D, w)}{\partial \vec{x}_n^*} = 0, \qquad \frac{\partial \chi^2(\vec{x}_0; D^*, w^*)}{\partial D^*} = 0$$

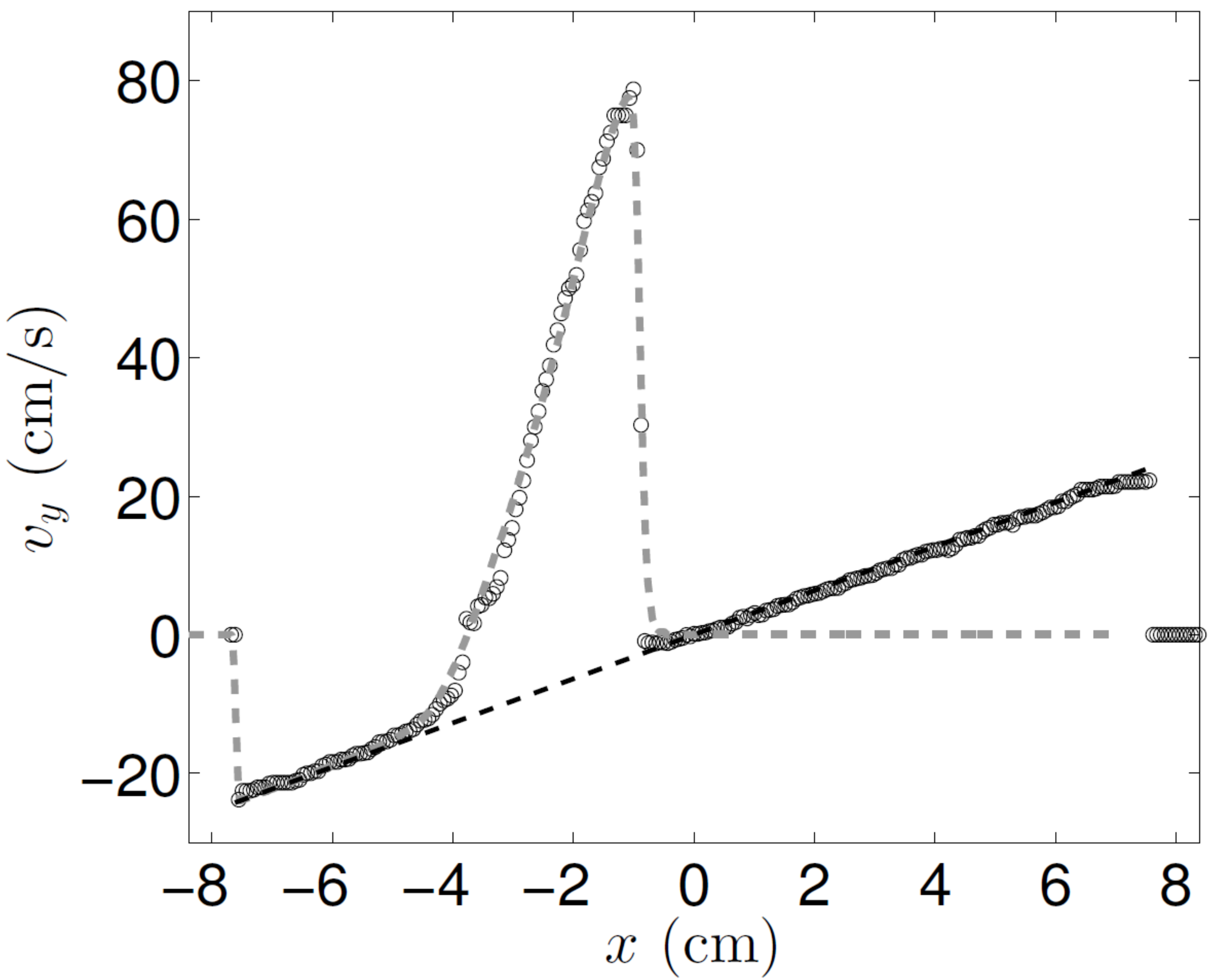
$$\frac{\partial \chi^2(\vec{x}_0; D^*, w^*)}{\partial w^*} = 0$$

$\chi^2 = 937.4$ $\chi^2 = 615.2$ $\chi^2 = 180.0$

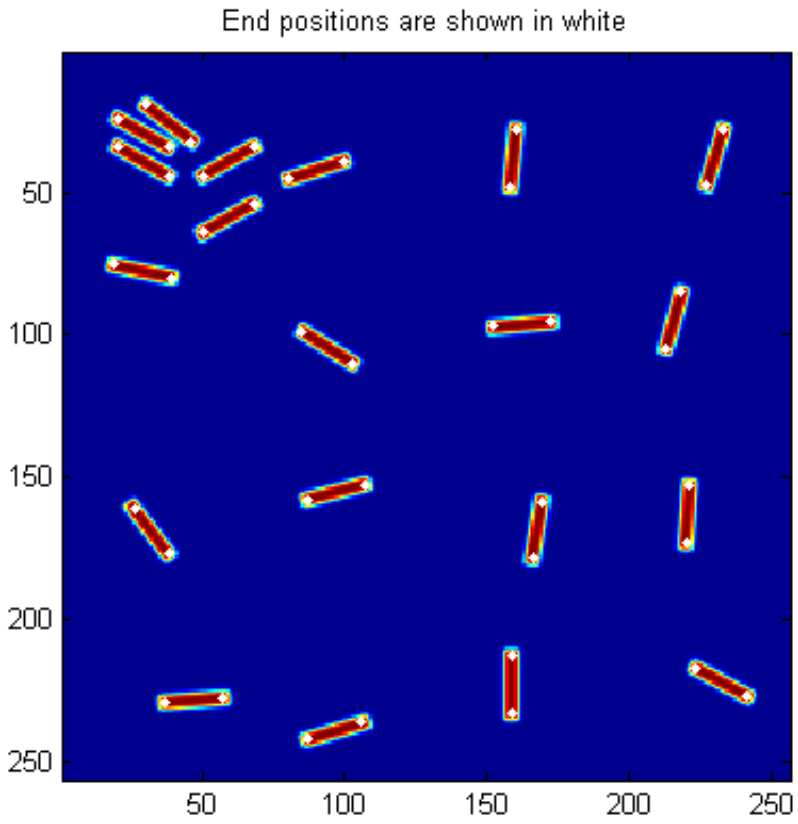


Original Positions Diameter-width

Particle Tracking vs. PIV



Particle Tracking
Extensions (shape)



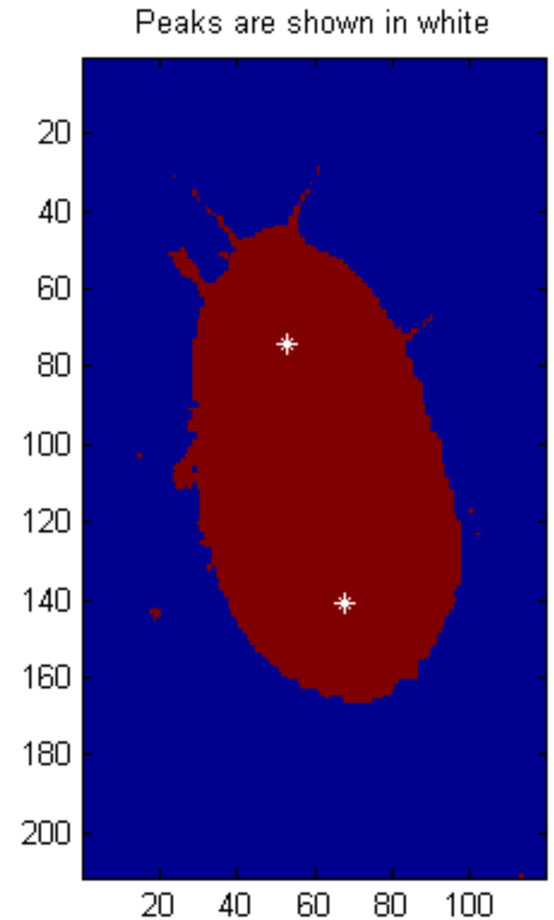
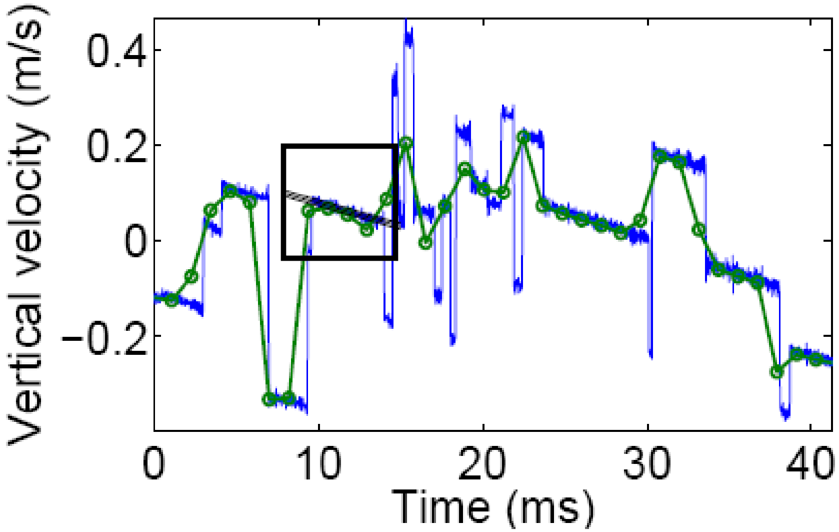


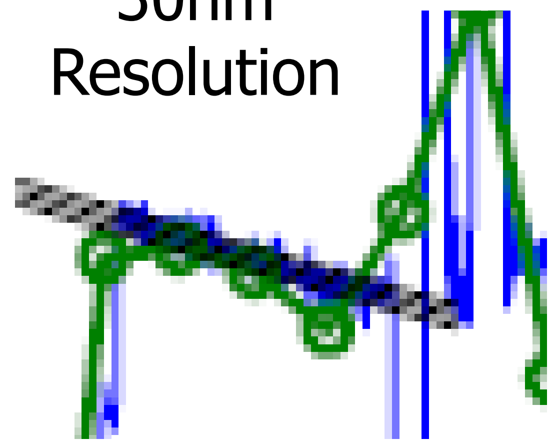
Figure 13.

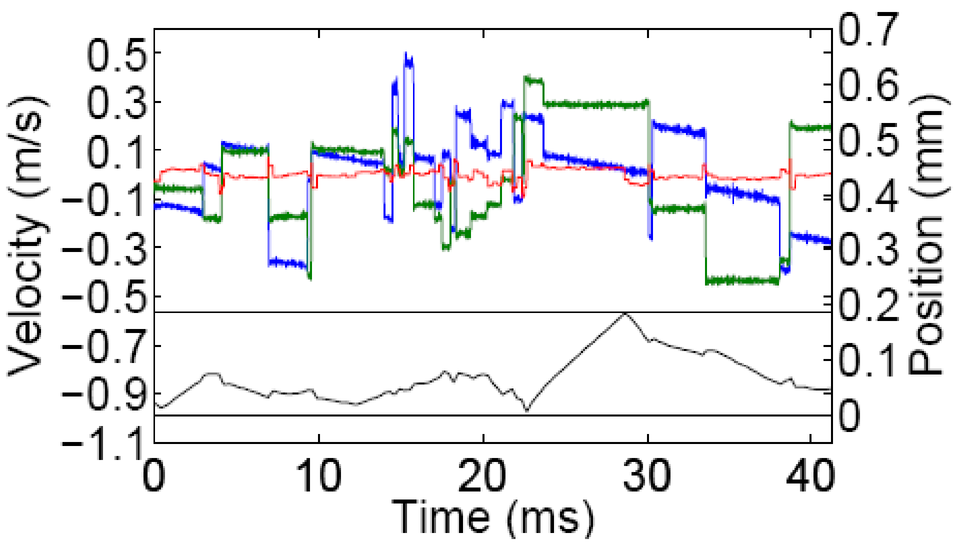
Particle Tracking

Extensions (High Accuracy, 3D)



30nm
Resolution

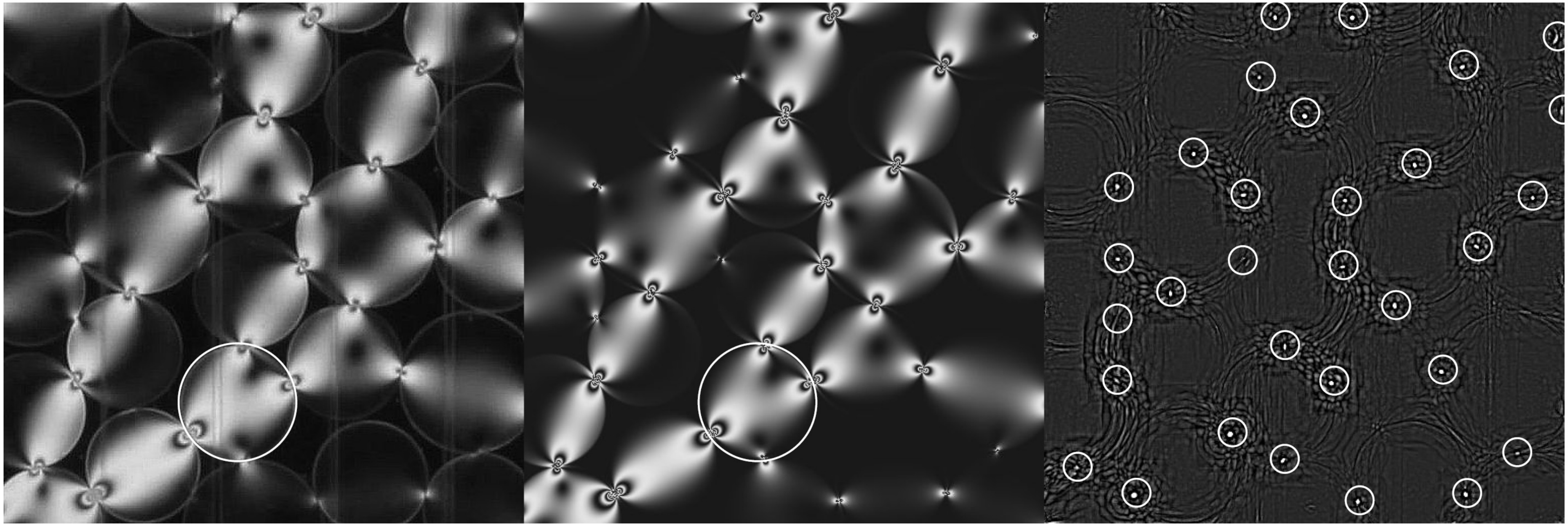




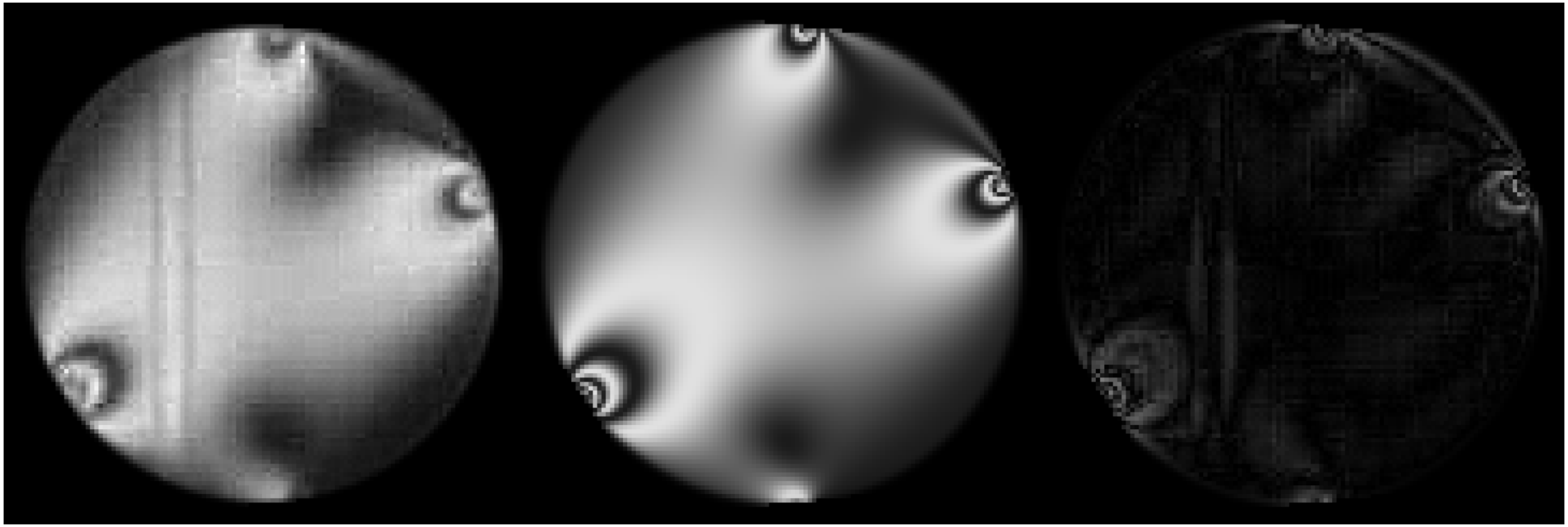
30nm/3mm =

$1/10^5$

Particle Tracking
Extensions (Force)

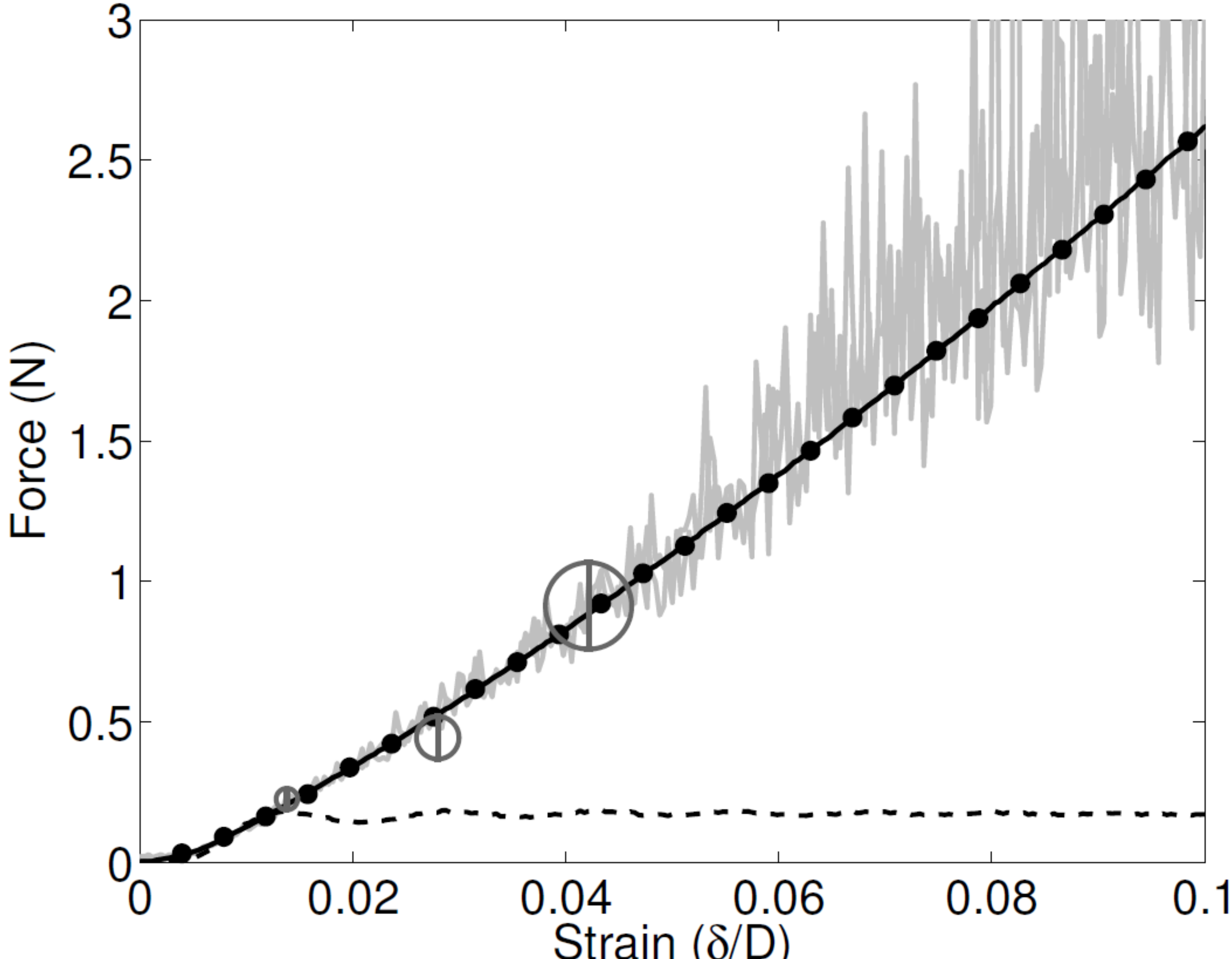


Particle Tracking
Extensions (Force)



Particle Tracking

Extensions (Force)



The End