Lecture 3 Image Sampling

COMP3204 & COMP6223 Computer Vision

How is an image sampled and what does it imply?

Book
pp
49-71

Department of Electronics and Computer Science

UNIVERSITY OF Southampton
School of Electronics and Computer Science

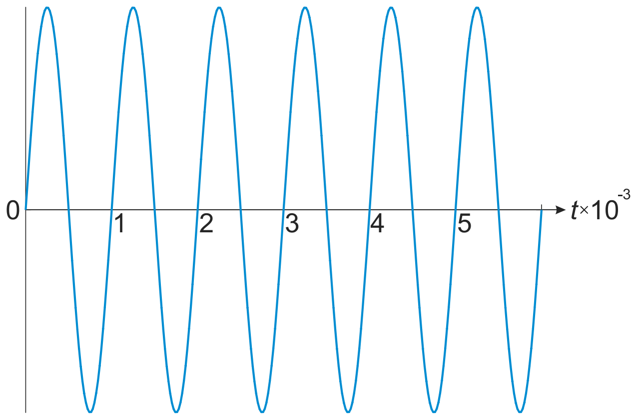
Aliasing in Sampled Imagery

(a) high resolution

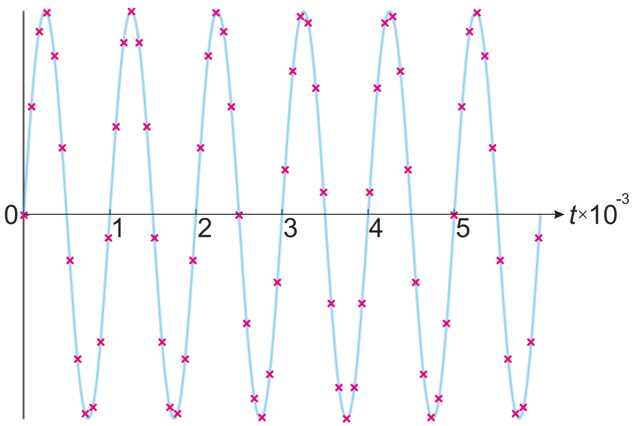
(c) low resolution – aliased

Sampling Signals

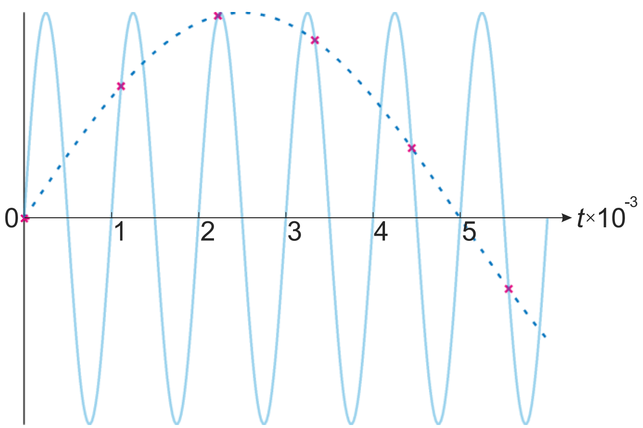
original signal



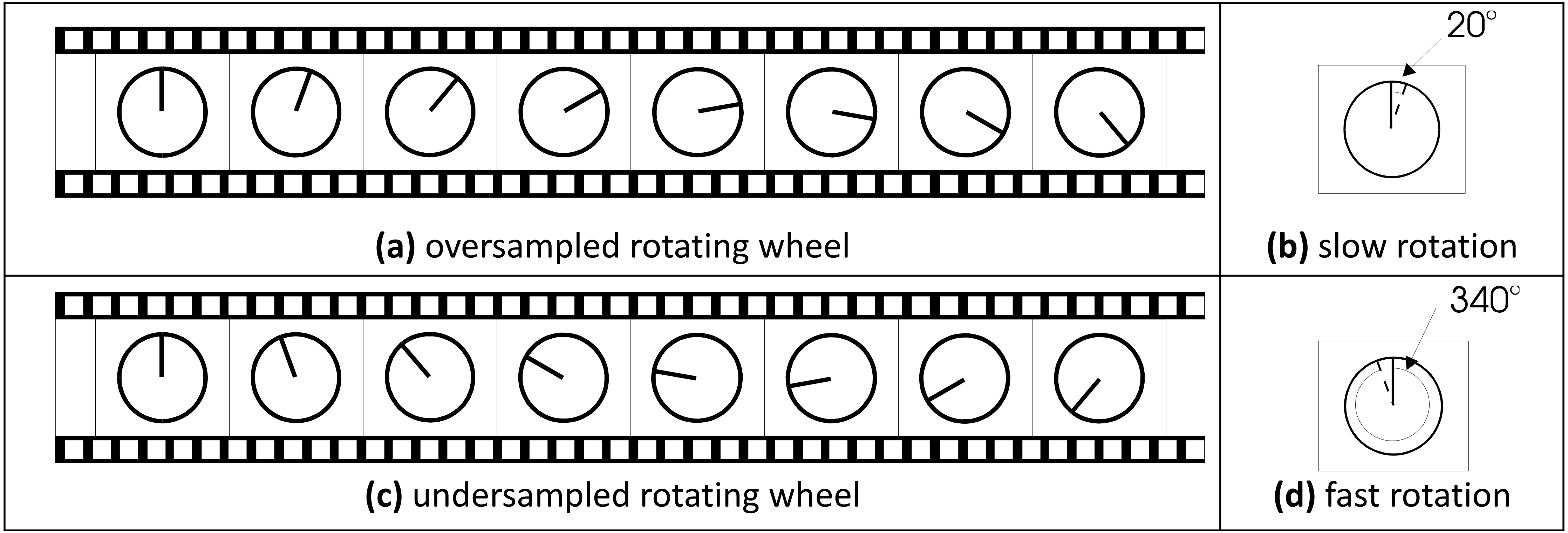
good sampling



bad sampling (aliasing)



Correct and Incorrect Apparent Wheel Motion



(a) oversampled rotating wheel

(b) slow rotation

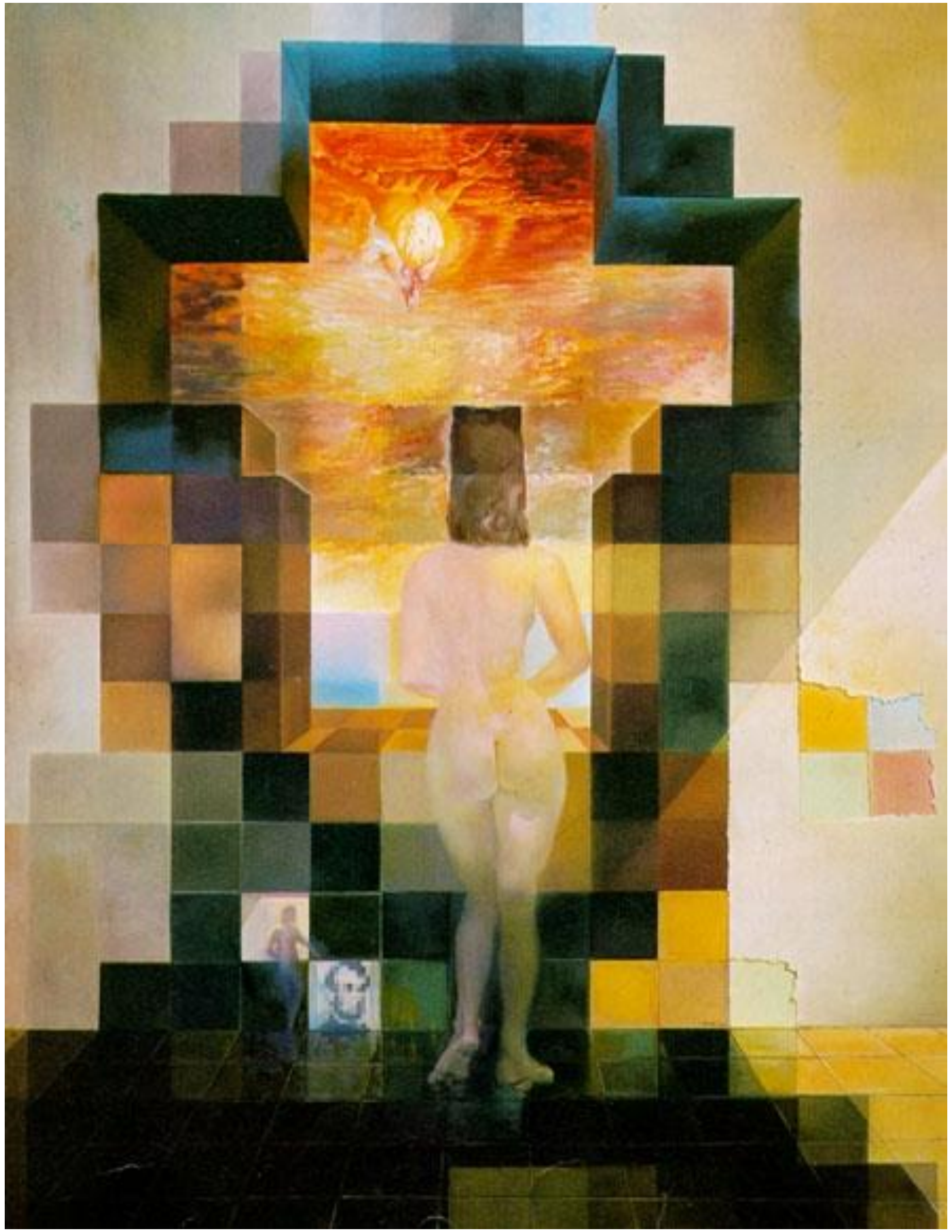
(c) undersampled rotating wheel

(d) fast rotation

https://www.youtube.com/watch?v=e1EqXE06xr8

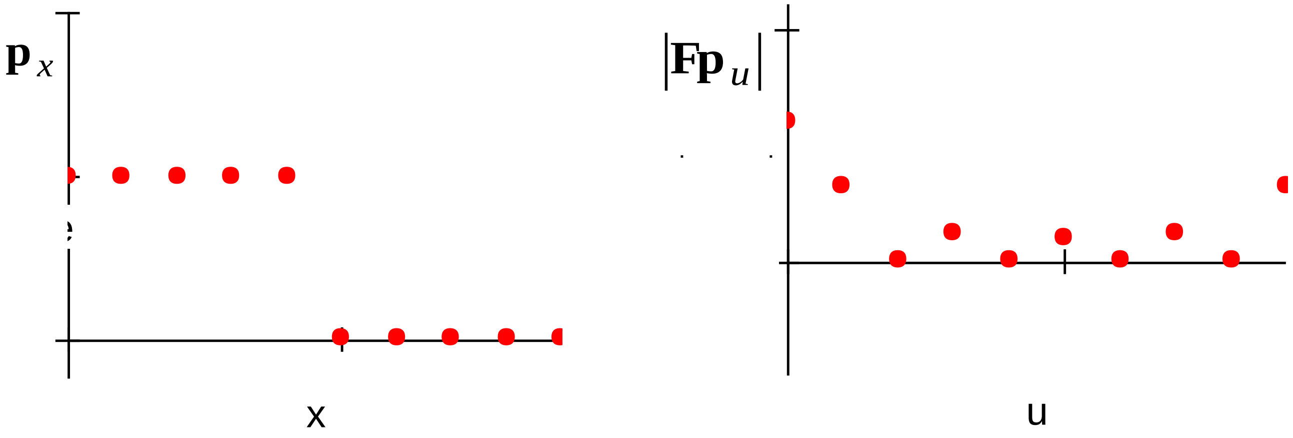
Sampling theory

- Nyquist’s sampling theorem is 1D
- E.g. speech 6kHz, sample at 12 kHz
- Video bandwidth (CCIR) is 5MHz
- Sampling at 10MHz gave 576×576 images
- Guideline: “two pixels for every pixel of interest”

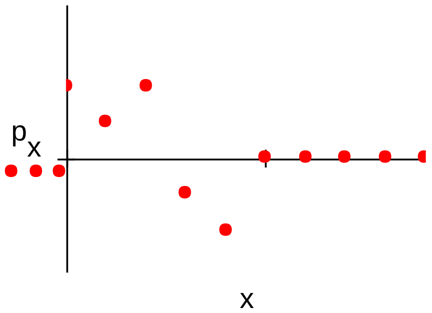


https://www.pinterest.com/pin/275423333431517864/

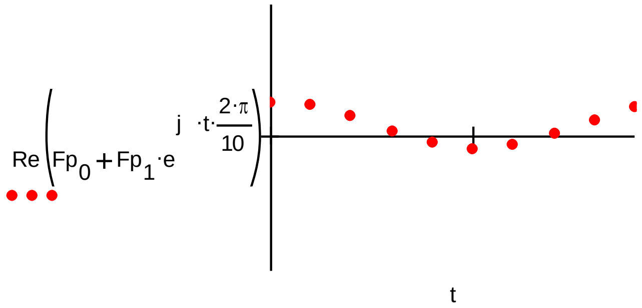
Transform Pair for Sampled Pulse



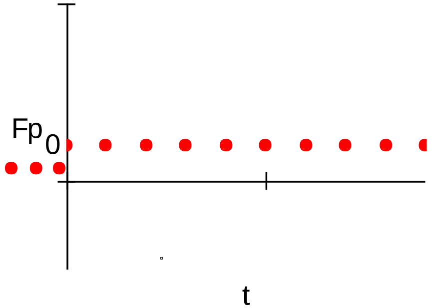
(a) sampled pulse (b) DFT of sampled pulse



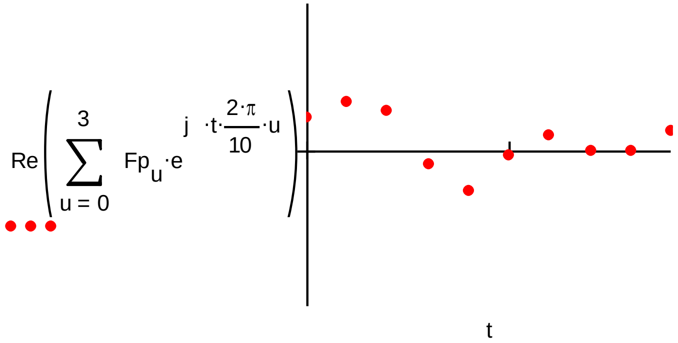
(a) original sampled signal



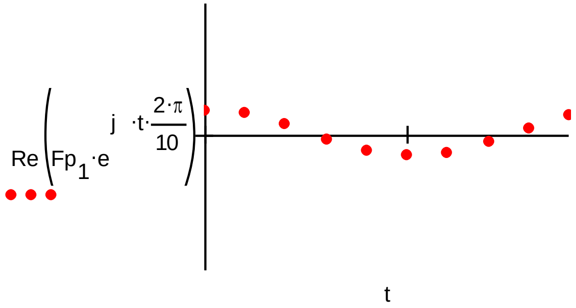
(b) first coefficient Fp_0



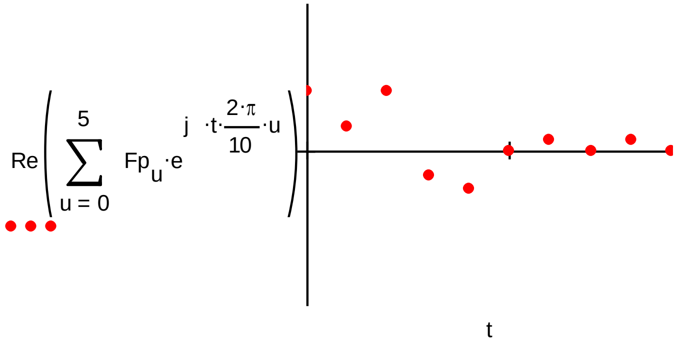
(c) second coefficient Fp_1



(d) adding Fp_1 and Fp_0



(e) adding Fp_0, Fp_1 , Fp_2 and Fp_3



(f) adding all six frequency components

Signal Reconstruction from Transform Components

2D Fourier transform

Forward transform $$\mathbf{FP}_{u,v} = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\mathbf{P}_{x,y}\, e^{-j\left(\frac{2\pi}{N}\right)(ux+vy)}$$

Where two dimensions of space, *x* and *y*

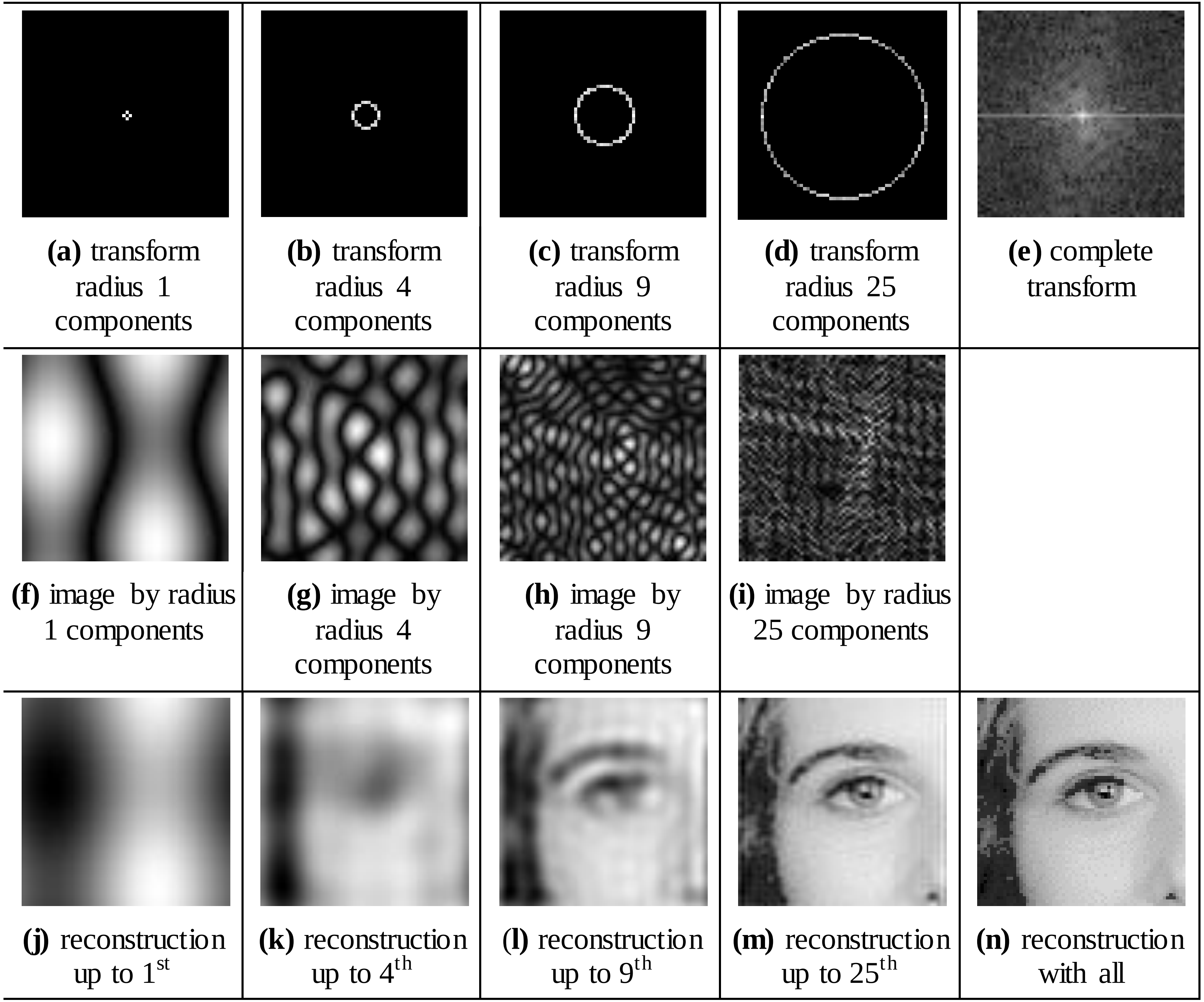
two dimensions of frequency, *u* and *v*

image *N*x*N* pixels $\mathbf{P}_{x,y}$

Inverse transform $$\mathbf{P}_{x,y} = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\mathbf{FP}_{u,v}\, e^{j\left(\frac{2\pi}{N}\right)(ux+vy)}$$

π??

Reconstruction

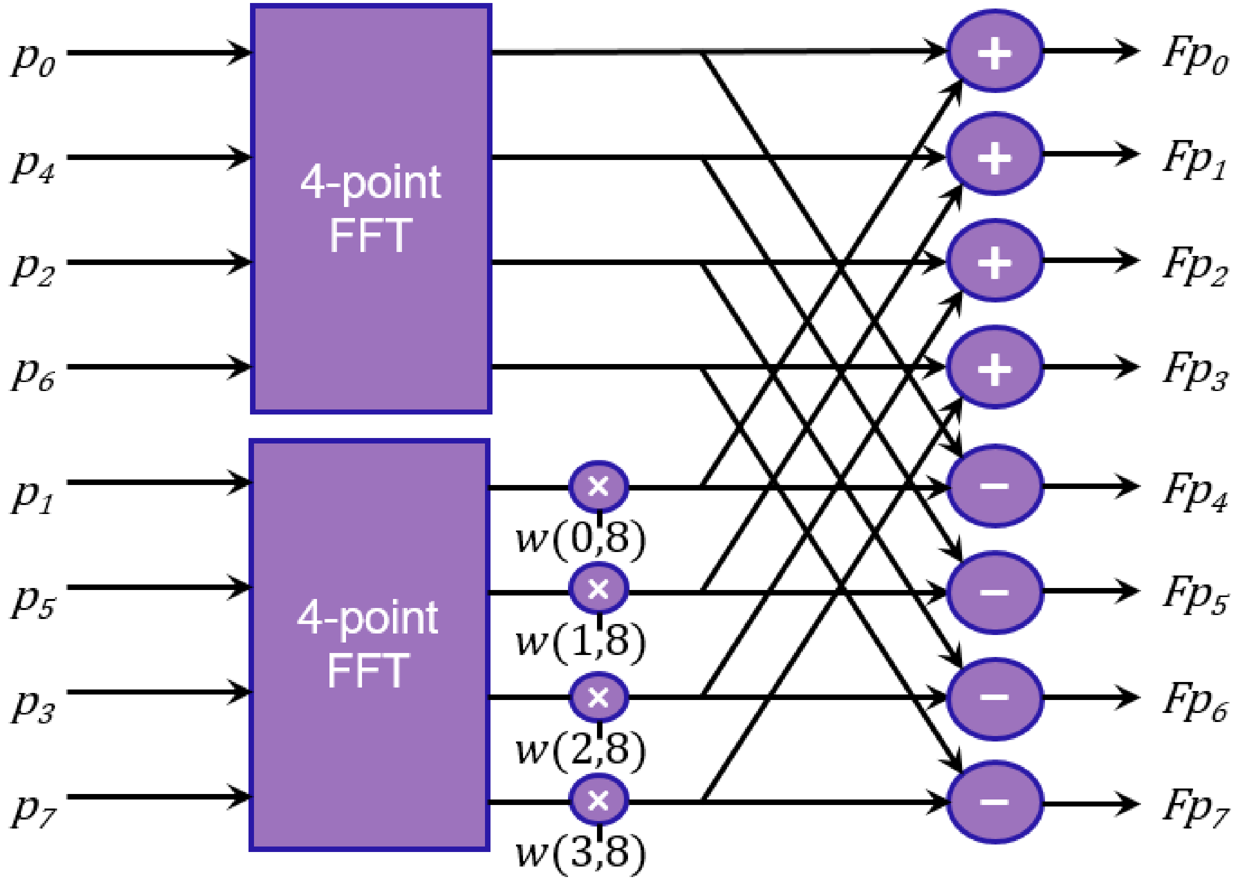


(a) transform radius 1 components
(b) transform radius 4 components
(c) transform radius 9 components
(d) transform radius 25 components
(e) complete transform

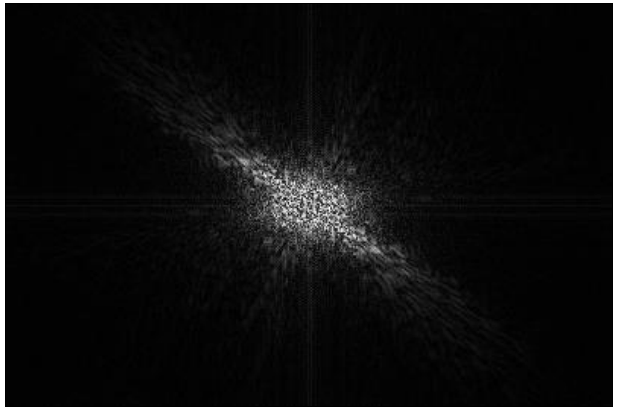
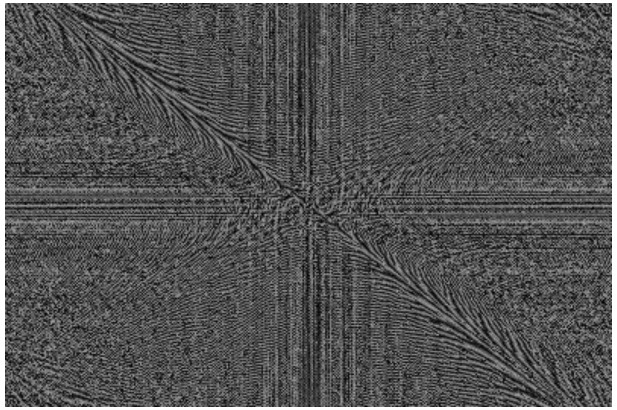
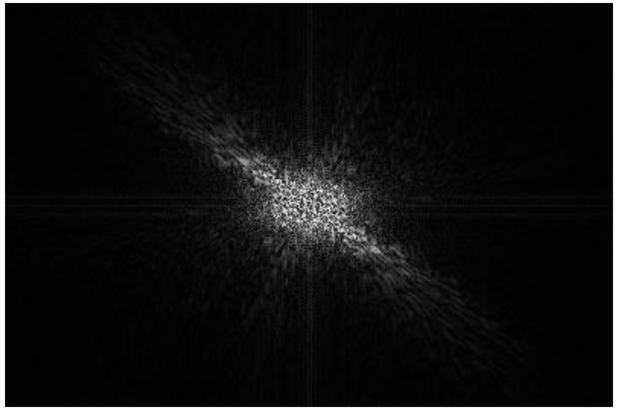
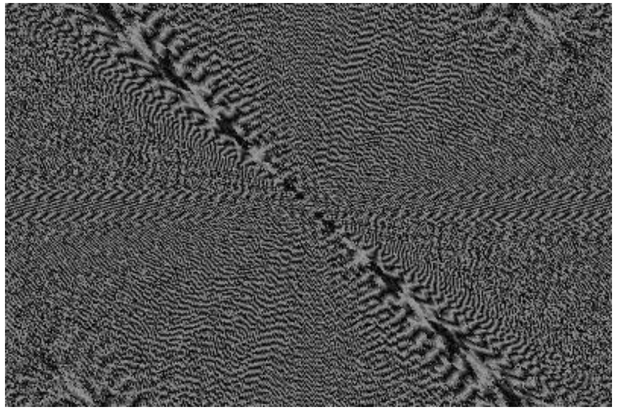
(f) image by radius 1 components
(g) image by radius 4 components
(h) image by radius 9 components
(i) image by radius 25 components

(j) reconstruction up to 1st
(k) reconstruction up to 4th
(l) reconstruction up to 9th
(m) reconstruction up to 25th
(n) reconstruction with all

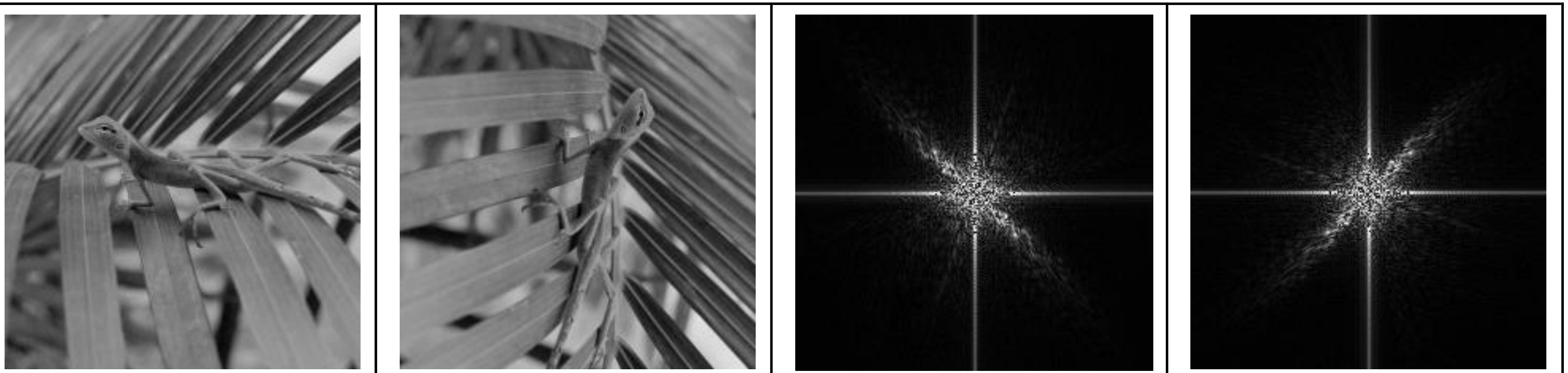
Implementation is via (Fast) FFT



Shift invariance

		
(a) original image	**(b)** magnitude of Fourier transform of original image	**(c)** phase of Fourier transform of original image
		
(d) shifted image	**(e)** magnitude of Fourier transform of shifted image	**(f)** phase of Fourier transform of shifted image

Rotation

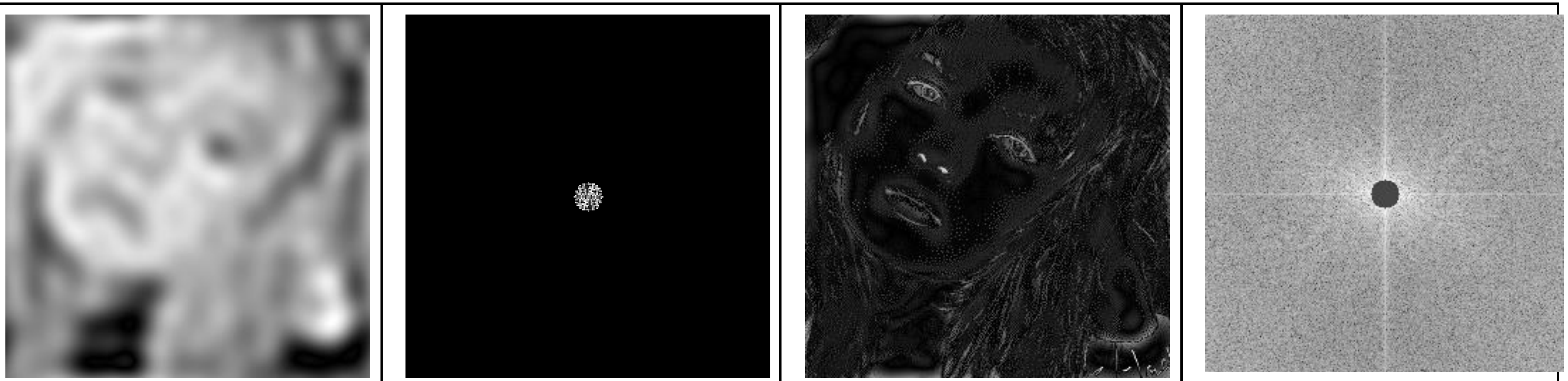


(a) original image	(b) rotated image	(c) transform of original image	(d) transform of rotated image

$$\mathbf{FP}_{u,v} = \frac{1}{N}\sum_{y=0}^{N-1}\sum_{x=0}^{N-1}\mathbf{P}_{x,y}e^{-j\left(\frac{2\pi}{N}\right)(uy+vx)}$$

Filtering

Fourier gives access to frequency components



(a) low-pass filtered image

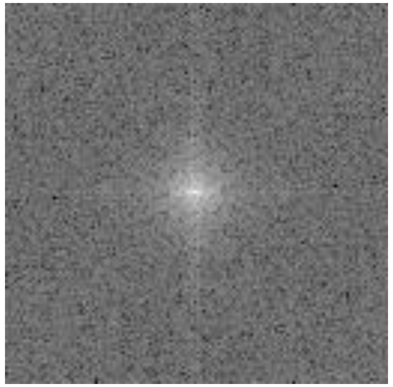
(b) low-pass filtered transform

(c) high-pass filtered image

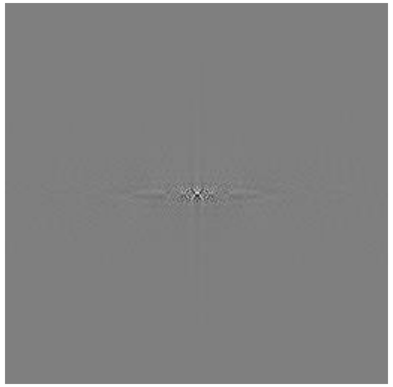
(d) high-pass filtered transform

Other transforms

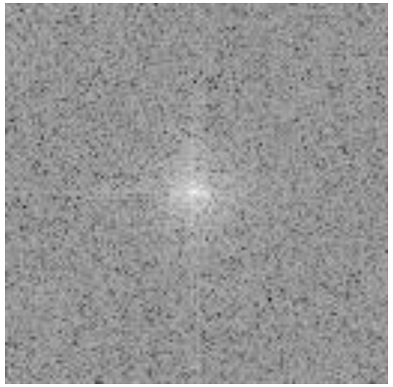
- For Lena



(a) Fourier transform



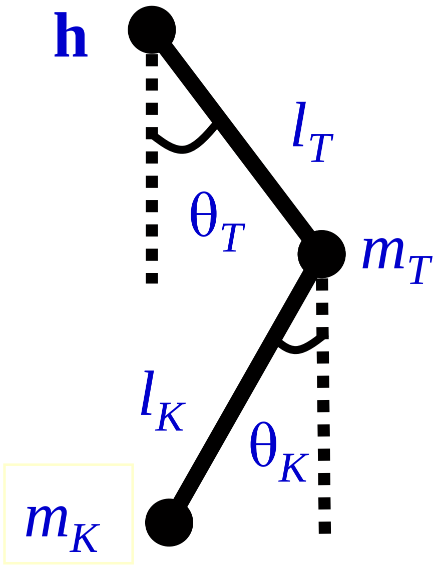
(b) discrete cosine transform

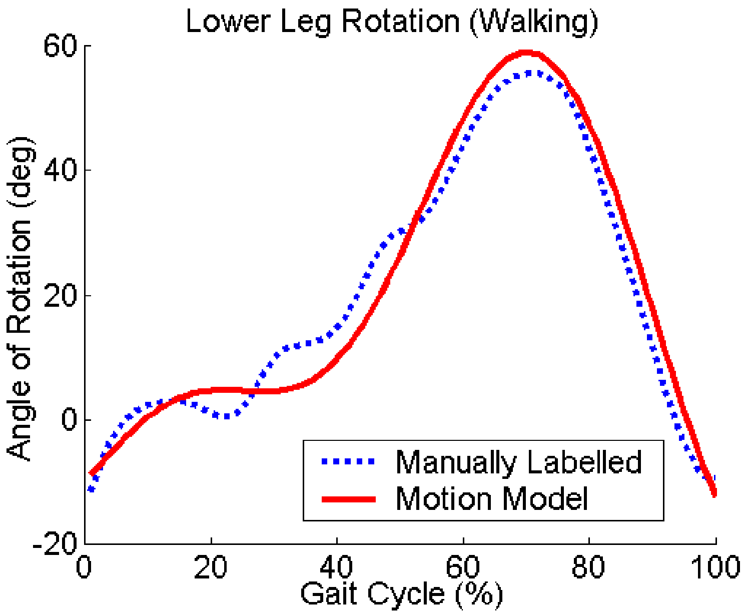


(c) Hartley transform

Modelling Gait(s)

- Extended pendular thigh-model, based on angles
- Uses forced oscillator/ bilateral symmetry/ phase coupling

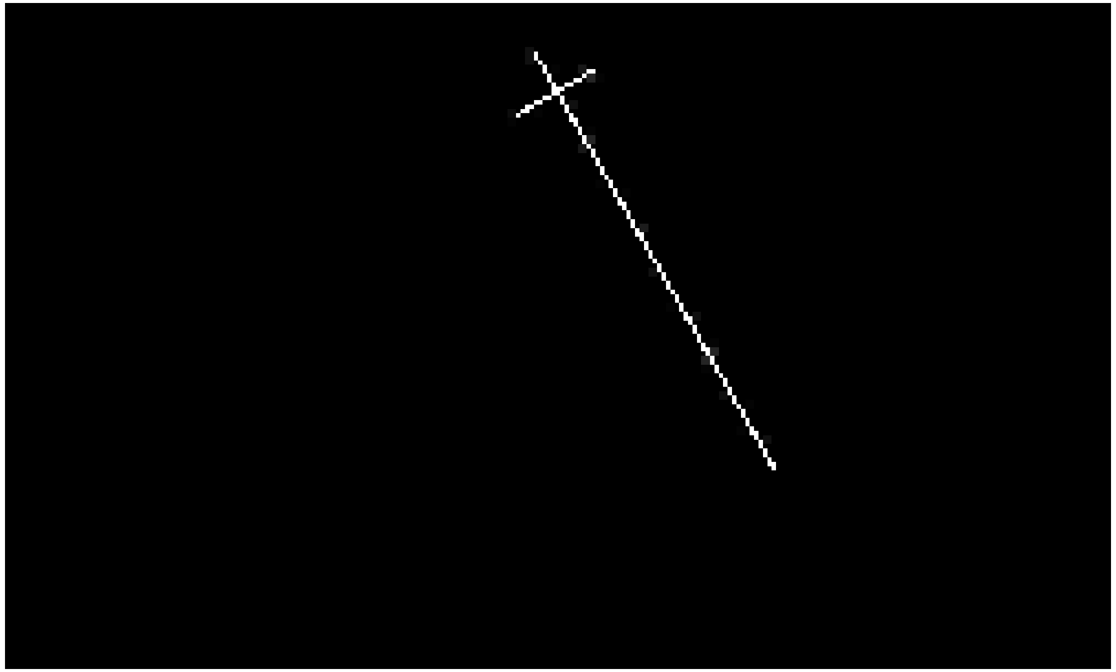




Yam, Nixon and Carter, *AVBPA* 2001, *Patt. Recog.* 2004

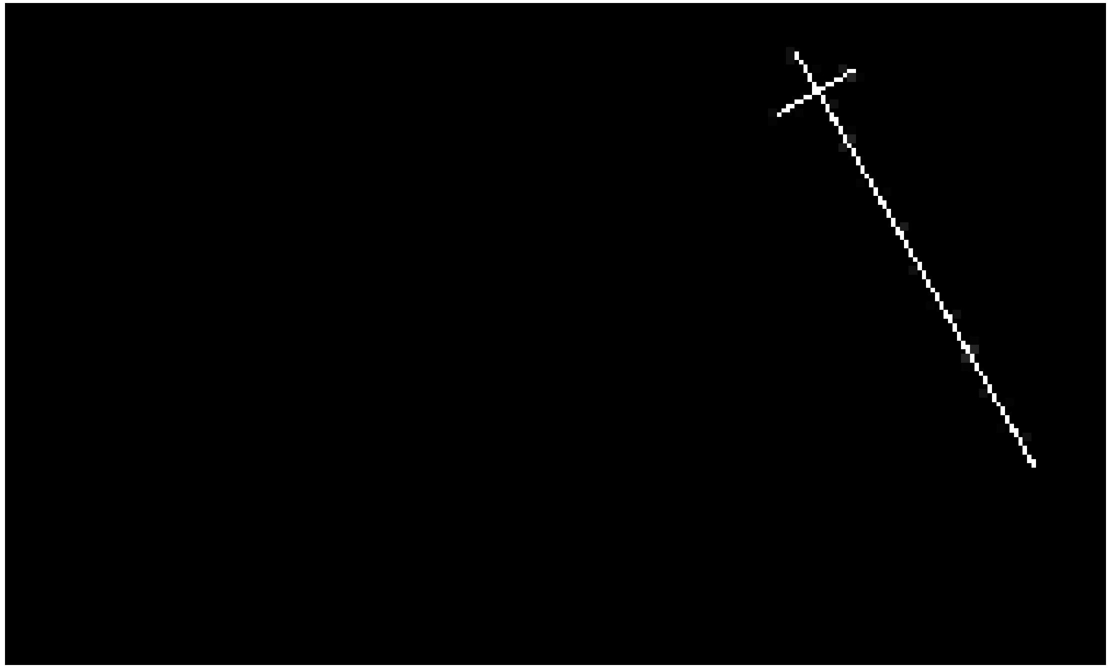
UNIVERSITY OF Southampton
School of Electronics and Computer Science

Modeling the Thigh's Motion 1



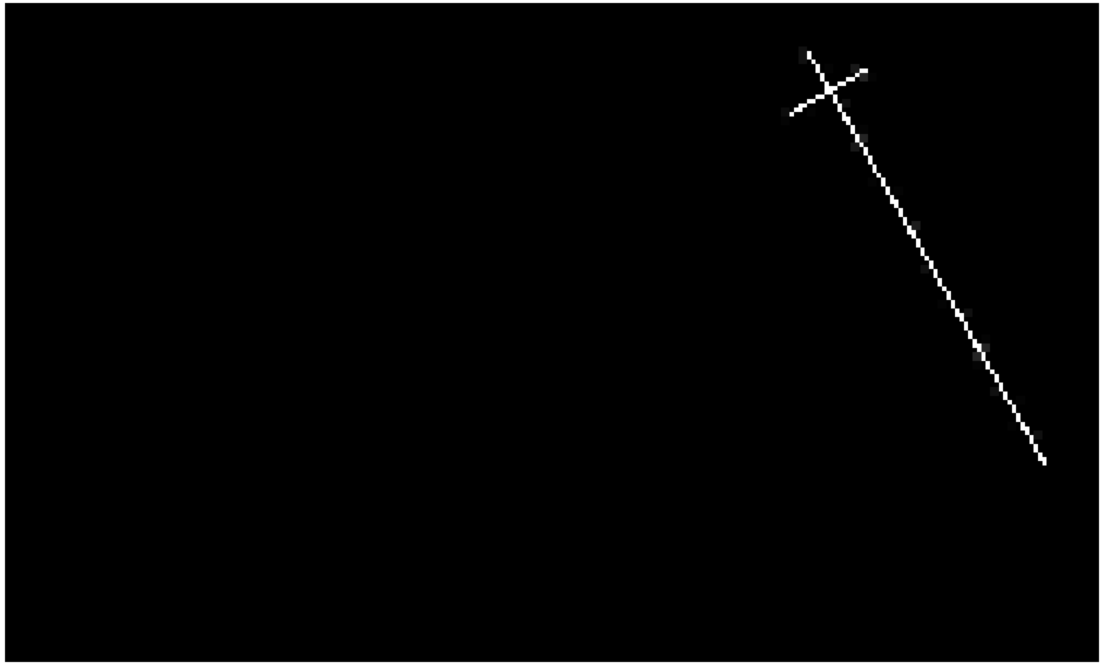
$$vs_x(t) = A\cos(\omega t + \phi)$$

Modeling the Thigh’s Motion 2



$$vh_x(t) = Vx + A\cos(\omega t + \phi)$$

Modeling the Thigh's Motion 3



$$\phi(t) = a_0 + \sum_{k=1}^{N} \left[b_k \cos\left(k\omega_0 t + \psi \right) \right]$$

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Validity?

Applications of 2D FT

- Understanding and analysis
- Speeding up algorithms
- Representation (invariance)
- Coding
- Recognition/ understanding (e.g. texture)