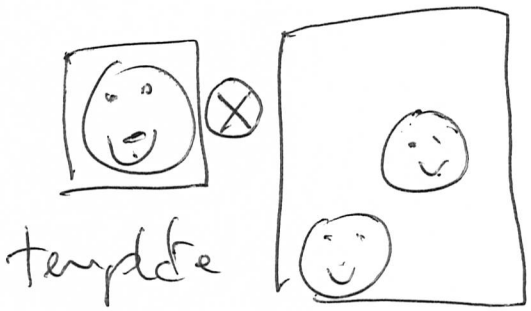
Finding shapes (220-231; 240-249)

i/. thresholding too simple - need shape

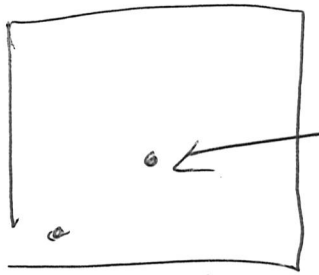
ii/. template matching.

store shape in a template

& match to image.



template



peak is location of template

accumulator

stores counts of matching points

optimal (for noise & occlusion) but v. slow

solution is to use fast Fourier

iii). correlation & convolution.

pic * temp. convolution $= F^{-1}(F(pic) \cdot \times F(template))$

(in spatial domain flip/invert template before convolution)

pic ⊗ temp. correlation $= F^{-1}(F(pic) \cdot \times (F(-template)))$

(matching) ↓ flipping process.

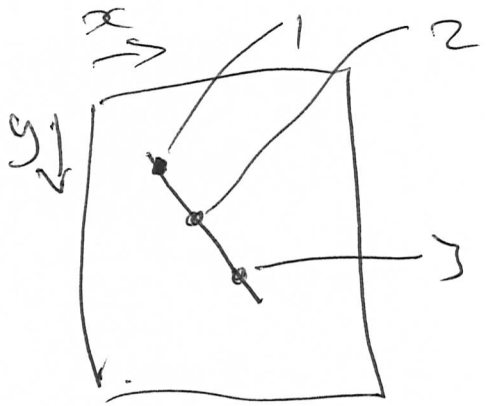
(in spatial domain no inversion of template)

flipping process

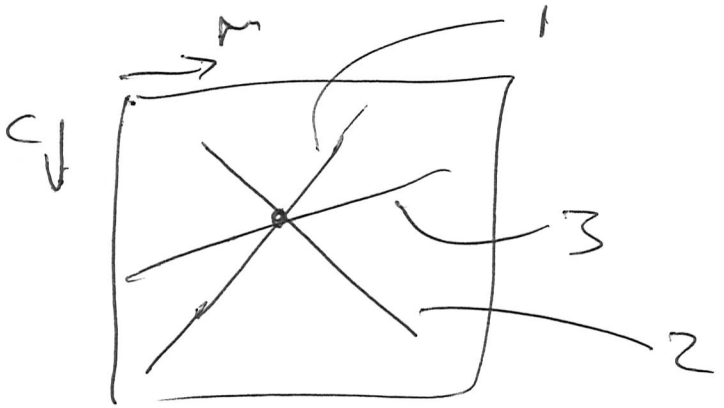
iv). Hough transform — same properties as template matching, but <u>faster</u>

$y = mx + c$ points (x, y) parameters (m, c)

$c = -xm + y$ points (c, m) " $(-x, y)$



image



accum

pseudo code

$\forall x, y$! search image

IF edge(x,y) > threshold ! is point significant

{ $\forall m \in m_{min} \dots m_{max}$

{ $c = -xm + y$

accum(m, c) PLUS 1 }}

afterwards argmax(accum) gives parameters