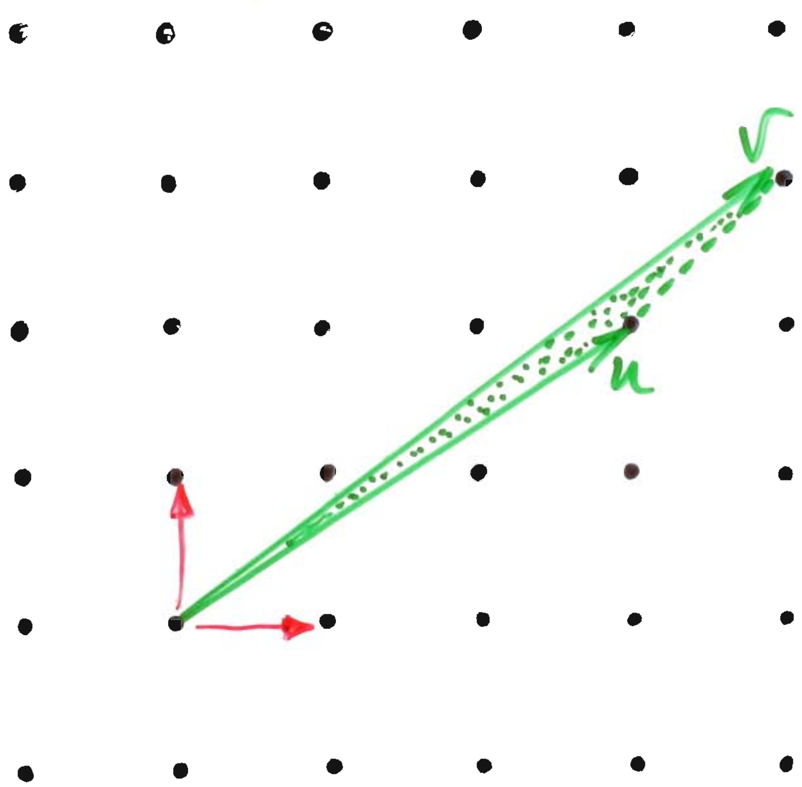
DIGITAL GEOMETRY for IMAGE-BASED METROLOGY

ALFRED M. BRUCKSTEIN

TECHNION

THE INTEGER GRID

(and other lattices in 2D)



The INTEGER grid is generated by

$$a \cdot [1,0] + b\,[0,1] = [a,b] \quad \text{where } a \in \mathbb{Z},\ b \in \mathbb{Z}$$

$\{[1,0], [0,1]\}$ a <u>BASIS</u>

Note that Area of $\{[0,0],[1,0],[0,1]\} = \frac{1}{2}$!

There are other BASES generating $\mathbb{Z}^2$!

$\underline{u}, \underline{v}$ generate $\mathbb{Z}^2$ iff

$$\{a\underline{u} + b\underline{v} \mid a, b \in \mathbb{Z}\} \equiv \mathbb{Z}^2$$

or the equation

$$a \cdot \underline{u} + b\underline{v} = [x, y]$$

has a solution in integers for every $[x,y] \in \mathbb{Z}^2$

$$\begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \frac{1}{u_1 v_2 - u_2 v_1} \begin{bmatrix} v_2 & -v_1 \\ -u_2 & u_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

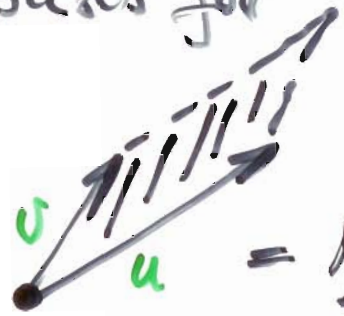
$\rightarrow$ If $u_1 v_2 - u_2 v_1 = \pm 1$ then $a \in \mathbb{Z}, b \in \mathbb{Z} \ \forall (x,y) \in \mathbb{Z}^2$

$\leftarrow$ If every $(x,y) \in \mathbb{Z}^2$ give $(a,b) \in \mathbb{Z}^2$ then

$\Delta = u_1 v_2 - u_2 v_1 \mid v_2, \ \Delta \mid u_2, \ \Delta \mid v_1, \ \Delta \mid u_1$

$\Rightarrow \Delta^2 \mid u_1 v_2 - u_2 v_1 \Rightarrow \Delta^2 \mid \Delta \Rightarrow \Delta = \pm 1.$

Hence $\underline{u}, \underline{v}$ are a basis for $\mathbb{Z}^2$ iff

$\left|\Delta \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix}\right| =$ Area  $= 1$

If $\{\underline{u}, \underline{v}\}$ form a basis then:

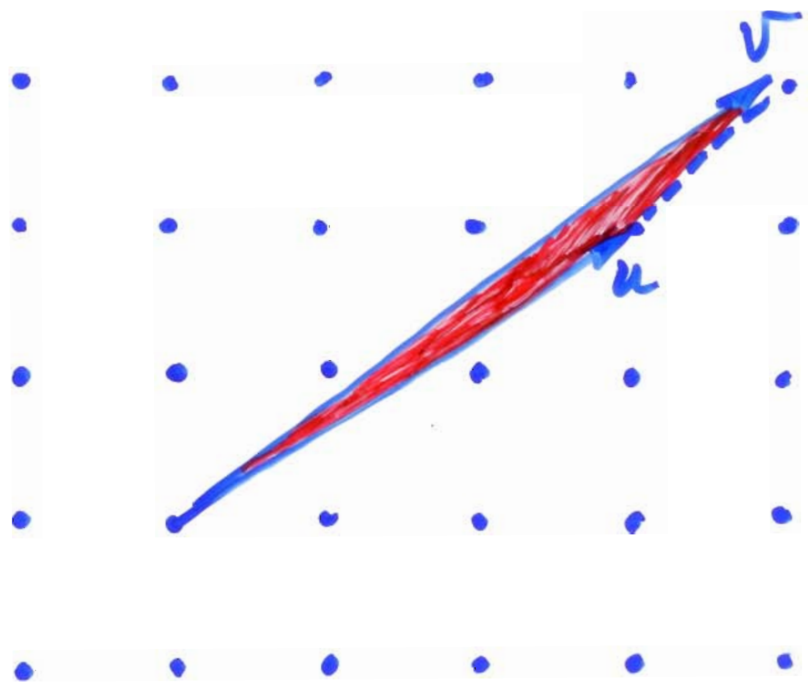
NO LATTICE POINTS EXIST IN The Area of the Rectangle 

Conversely: if NO POINTS EXIST inside the Triangle defined by $\underline{m}$ & $\underline{n}$ (two independent Lattice points) then $\{\underline{m}, \underline{n}\}$ form a BASIS for $\mathbb{Z}^2$.

THEOREM

Any elementary triangle with vertices at grid points and no grid points inside and on the edges 1. define a basis for $\mathbb{Z}^2$

2. has area $1/2$.



$\boxed{\text{Area} = 1/2}$

THEOREM (PICK)

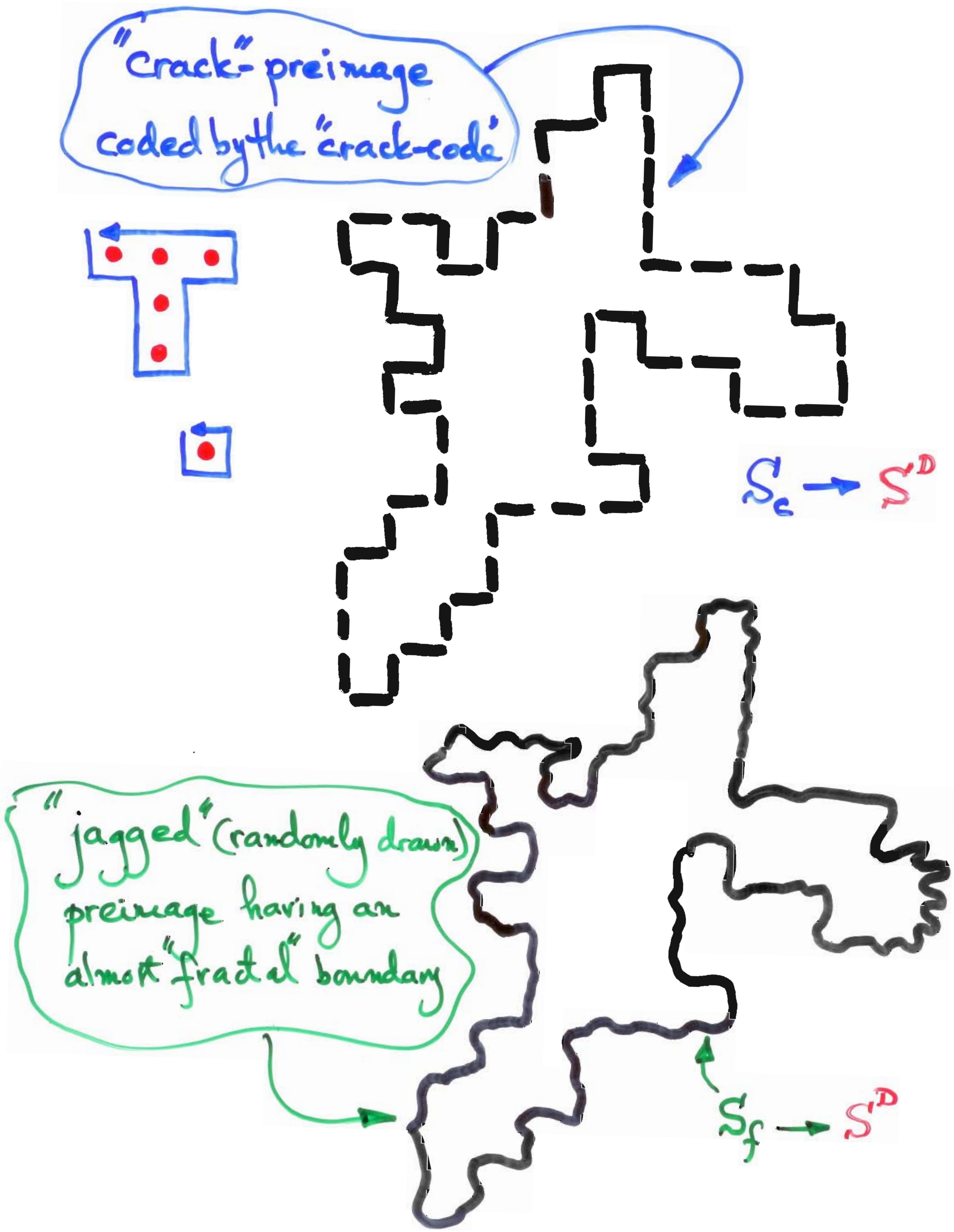
If P is a simple planar polygon
with all vertices at LATTICE POINTS
and it has B Lattice Points on the boundary
I Lattice Points inside

Then: $\text{Area}(P) = I + \frac{B}{2} - 1$.

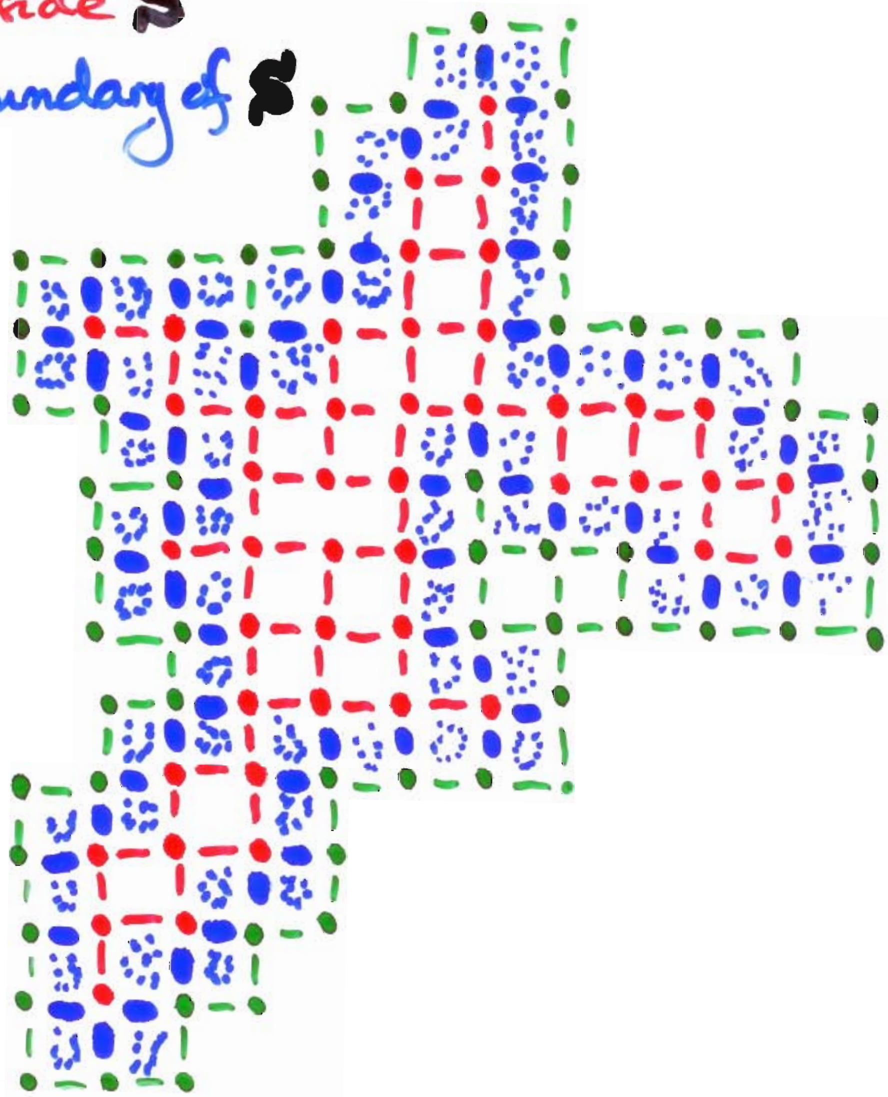
PROOF (from "the Book")
TRIANGULATE P; No of elementary
triangles is $B + 2I - 2$ and all have
area $1/2$ hence

$$\text{Area}(P) = \frac{1}{2}(B + 2I - 2) = I + \frac{B}{2} - 1.$$

Possible Preimages of S^D

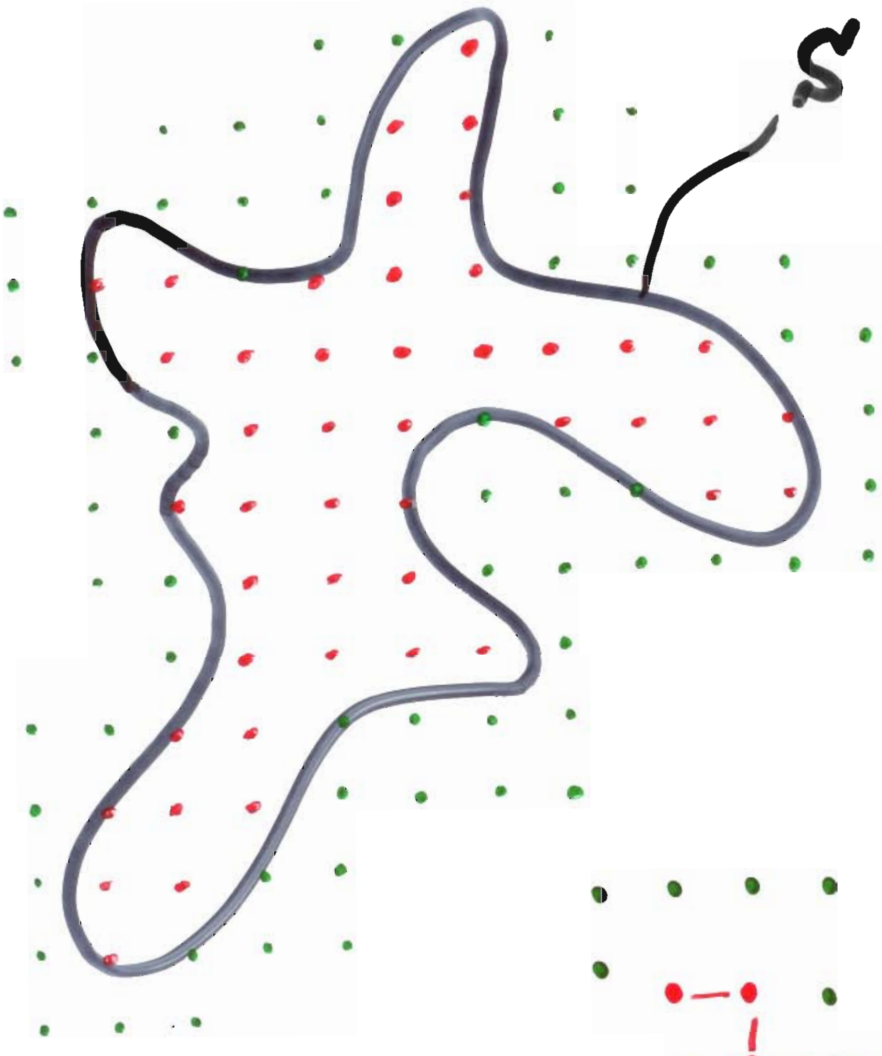


green: outside S
red: inside S
blue: boundary of S



The mapping from continuous shapes to their "digitized" representations is "many-to-one" so we must make assumptions about the shape S w.r.t. the sampling grid.
Many "preimages" that digitize to S^D can be considered.

THE PROBLEM



Sample S on a grid.

From sampled image of S learn as much as possible on S.

WHAT DO WE WANT TO KNOW ABOUT S FROM S^D?

- Location Information
  (where is the centroid of S)

- Geometric Measures
  Area, Perimeter, Moments (orientation)

- Does S fit some specifications
  design
  (is it a circular shape, a triangle, a half-plane etc..)

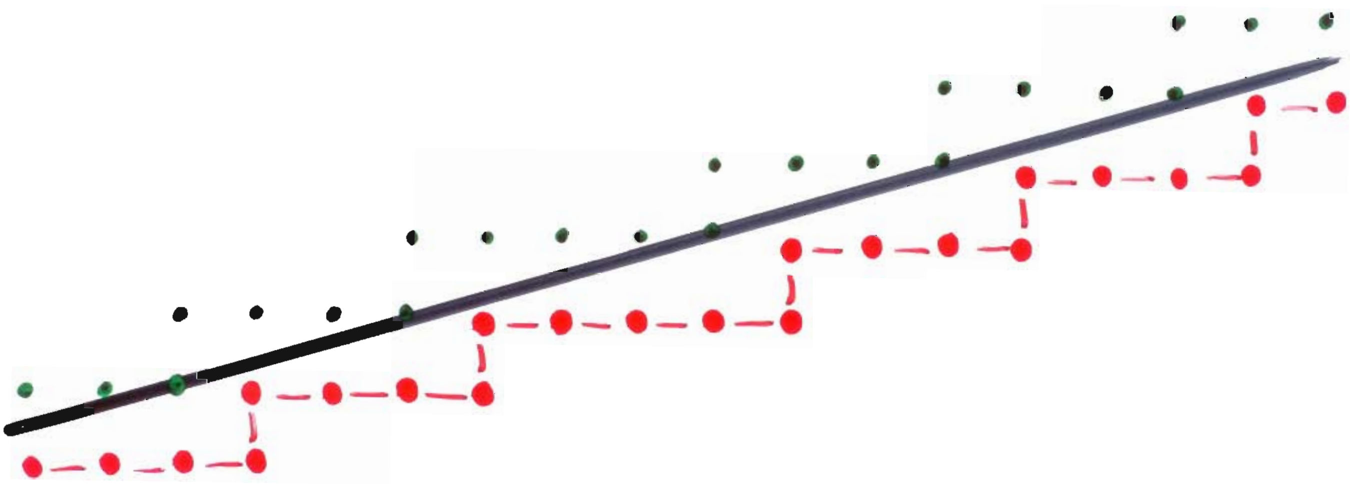
- Is S made of several components of given sizes and locations? ...

Applications: particle analysis, medical specimen testing, PCB-inspection, VLSI metrology mechanichal part-quality control

THE "ROYAL" EXAMPLE

- DIGITAL STRAIGHT LINES / BOUNDARIES

S ~ a half-plane is digitized by point sampling



Crack-code:

... RRRU RRRU RRRRU RRRU RRRUR ...

Questions
- characterize properties of digital line crack codes
- detect efficiently crack-code segments with linear preimages
- locate linear boundaries & estimate slopes

THE SELF-SIMILARITY OF DIGITAL STRAIGHT LINES

... RRRU RRRU RRRRU RRRU RRRU ...

... RRU RRU RRRU RRU RRU ...

... RU RU RRU RU RU ...

... U U R U U U ...

(exchange U→R, R→U)

... R R U R R R ...

... R U R R R ...

and so on... and on... and on...

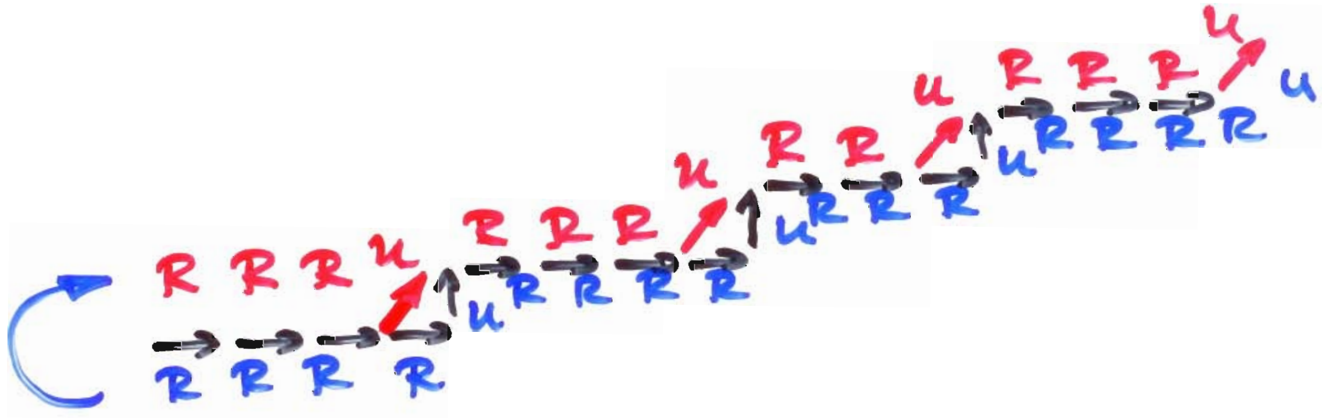
THEOREM:

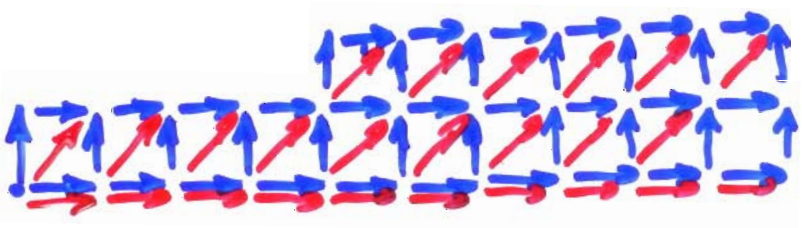
$\boxed{\text{Such transformations map digital straight line codes into digital straight line codes.}}$

i.e. transformed sequence is DSL iff original sequence is DSL.

BEAUTIFUL PROOF: New sequences are re-encodings of the boundary on EMBEDDED LATTICES.

SEEING THE PROOF:





old basis generating $\mathbb{Z}^2$

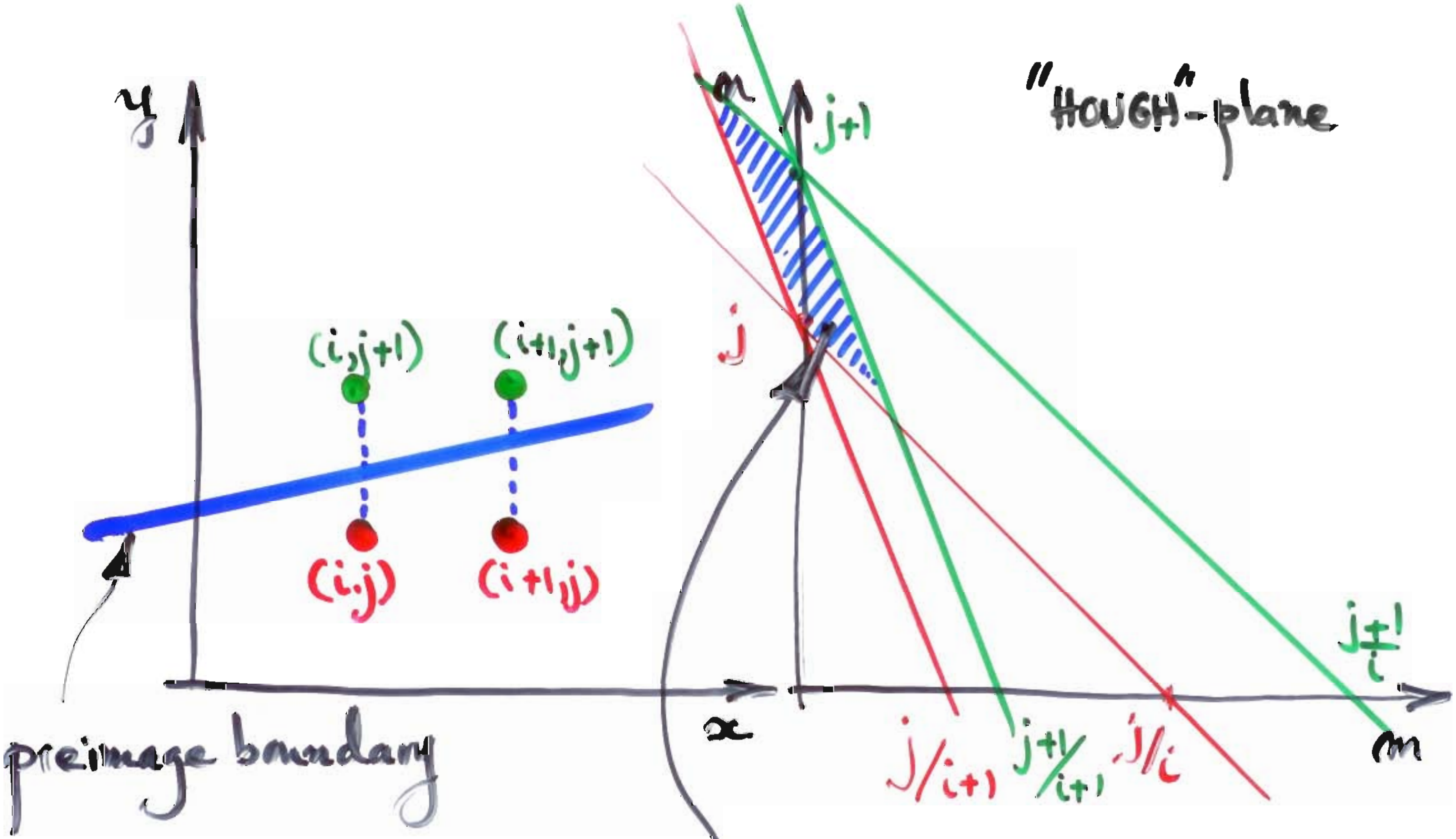
new basis generating $\mathbb{Z}^2$

SIMPLE OBSERVATION:

THE SAME LINEARLY SEPARABLE DICHOTOMY OF $\mathbb{Z}^2$ CAN BE CRACK-CODED IN (BY) ALL POSSIBLE BASES OF $\mathbb{Z}^2$!

BASIS TRANSFORMATIONS ARE 2×2 MATRICES REPRESENTING $GL(2,\mathbb{Z})$.

DETECTING DIGITAL STRAIGHT LINE SEGMENTS



all lines that cut the segments $[(i,j)\,(i,j+1)]$ and $[(i+1,j)\,(i+1,j+1)]$

$$y = mx + n$$

$$\begin{cases} j = mi + n \\ j = m(i+1) + n \end{cases} \qquad \begin{cases} j+1 = mi + n \\ j+1 = m(i+1) + n \end{cases}$$

ALGORITHM: For each symbol of the crack-code cut the "uncertainty region" with the corresponding band in the Hough-plane. While the result is not empty $\exists$ a linear preimage.

COMPLEXITY of the ALGORITHM:

- at each step the "Uncertainty Region" or the "LOCALE" of the DSS is a polygon of at most 4 sides (the Dorst-miraculous (McIlroy) result!)
- the locale is encoded by about 8 integers
- the cut takes $O(1)$-operations and results in a new locale or in EMPTY

THEREFORE WE HAVE AN $O(1)$/STEP

RECURSIVE SIMPLE DSS

DETECTION PROCEDURE

Note that the locale provides estimates for the position and slope of the digital straight segment.

Hence we can readily use this algorithm for parsing chain codes into straight segments and DO LENGTH ESTIMATION TOO!

PERIMETER ESTIMATION

- Given a (chain-coded) boundary estimate the perimeter of the preimage

estimate perimeter by length of green line

Procedure: Classify crack-code symbols/portions of size L/ according to configuration in the neighborhood, then each symbol type can be assigned a "weight" (length).

$$\text{PERIMETER} = \sum_i (\text{Class Weight}) \cdot \#\text{symbols of class } i$$

DESIGN PROCEDURE:

optimize weights to get "uniformly" good length estimates for digital-lines at all directions.

This may be regarded as a generalized Proffit-Rosen length estimation process.

BUT:

PARSING THE BOUNDARY INTO DIGITAL STRAIGHT SEGMENTS AND ADDING THE LENGTH OF THE STRAIGHT PREIMAGES

THEORETICALLY BEATS ALL OTHER LENGTH ESTIMATORS!

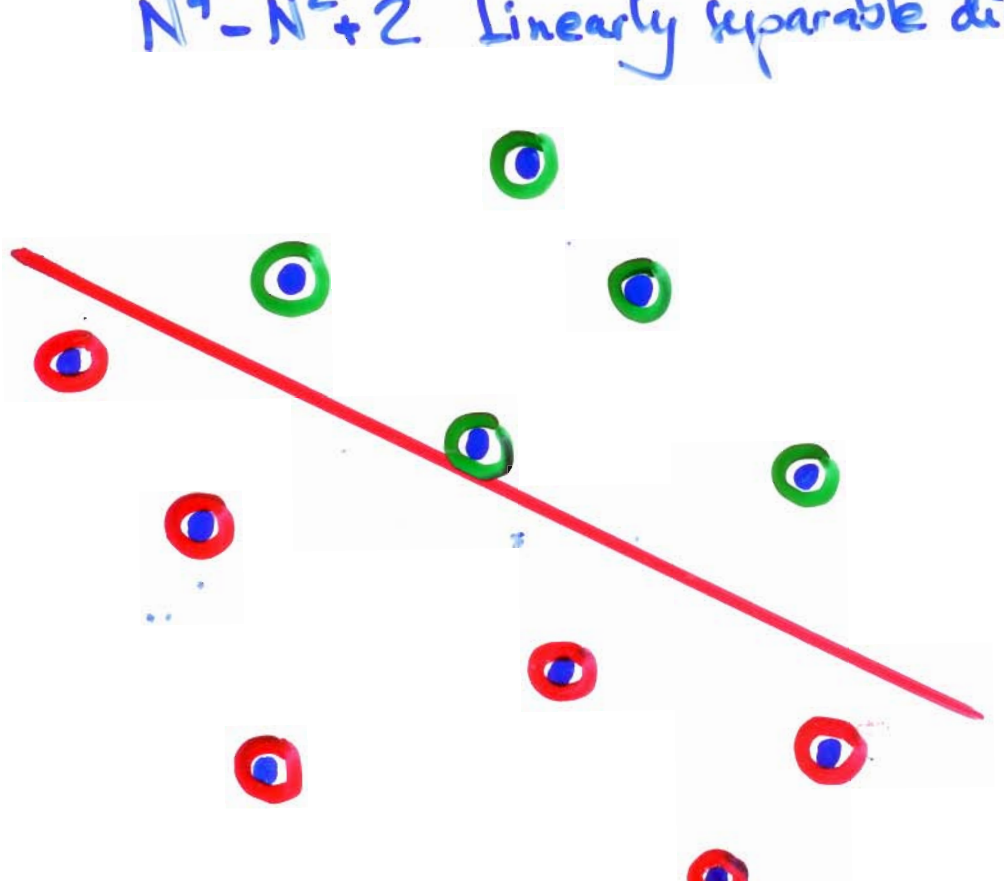
COUNTING LINEARLY SEPARABLE DICHOTOMIES of an NxN GRID

If we have K points in "general position" in the plane, there are

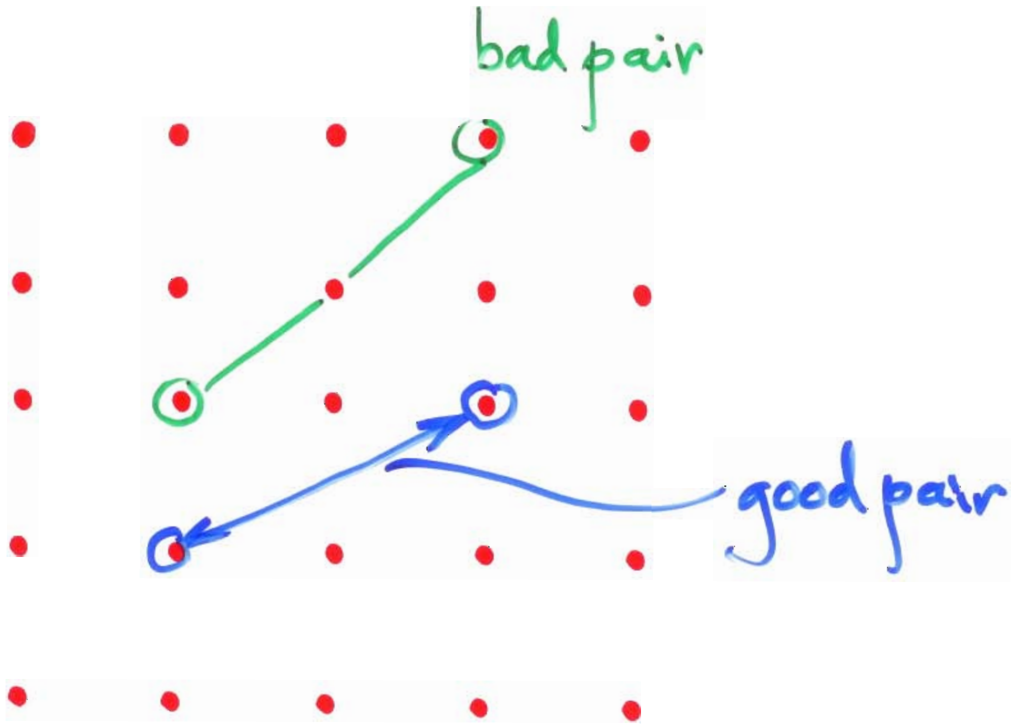
$$2\sum_{i=0}^{2}\binom{K-1}{i} = K^2-K+2 \text{ linearly separable dichotomies.}$$

So $K=N^2$ points in general position admit

N^4-N^2+2 Linearly separable dichotomies.



$\exists\ N^4-N^2+2$ DIGITAL LINES ON A TOTALLY IRREGULAR ARRAY OF POINT SENSORS



We must count here the number of pairs of grid points (i,j) & (k,l) so that the line between (i,j) & (k,l) does not pass through another grid point. Hence we need to have $(i-k)$ an $(j-l)$ relatively prime!

Two numbers chosen $\overset{\text{uniformly}}{\vee}$ between 1 and K are relatively prime with probability $6/\pi^2$.

$\boxed{\text{AMAZINGLY: choosing two pairs at random } (i,j) \text{ \& } (k,l) \text{ from the } N^4 \text{ possibilities we have that } (i-k) \text{ and } (j-l) \text{ will be relatively prime with the same probability } 6/\pi^2.}$

WHAT ARE WE COUNTING?

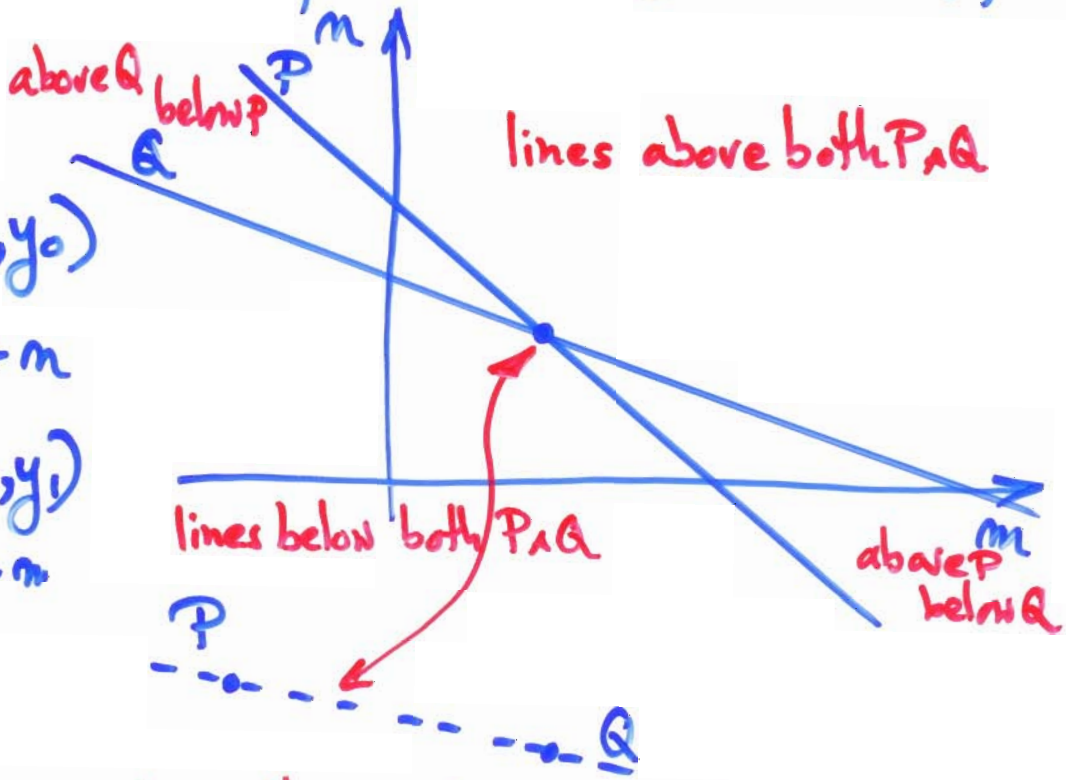
In the Hough-plane we determine all the lines that correspond to the given set of points.

$P = (x_0, y_0)$

$y_0 = m x_0 + n$

$Q = (x_1, y_1)$

$y_1 = m x_1 + n$



We are counting the # of regions defined in the Hough-space by a set of lines determined by the set of points.

WE HAVE TO DO THE SAME FOR POINTS ON THE GRID!

Disks, Convex & Star-Shaped Preimages

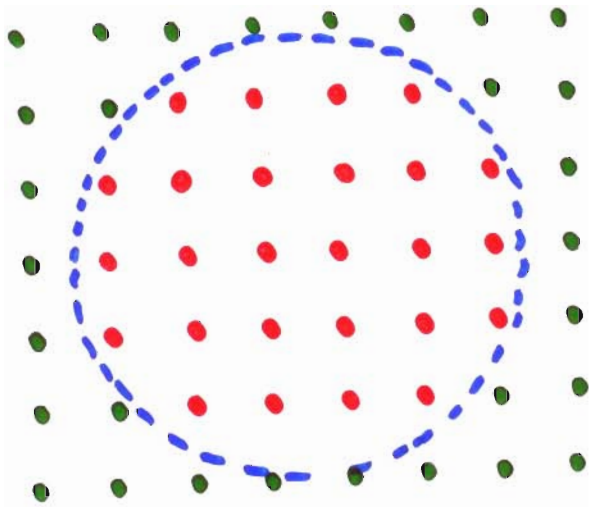
Suppose we want to test whether a discretized image S^D could be the image of a circular shape S (a solid black disk)

Similar questions to what we have seen in the context of half-planes arise.

- characterize digital disks
- determine their LOCALES
- find algorithms to determine "circular" segments
- count # disks on an $N \times N$ grid etc.

Some of these questions are not satisfactorily solved yet (to the best of my knowledge!).

THE "LOCALE" for a disk of unknown radius:



- THE LOCUS OF POINTS CLOSER TO all THE RED POINTS THAN TO ANY GREEN POINT

A WELL-KNOWN CONCEPT IN COMPUTATIONAL GEOMETRY

IT IS A CONVEX REGION THAT CAN BE EFFICIENTLY DETERMINED.

OF COURSE IF WE A-PRIORI KNOW THE RADIUS: R

WE CAN SIMPLY INTERSECT THE EXTERIORS OF ALL DISKS OF RADIUS R AROUND THE GREEN POINTS WITH ALL THE DISKS OF RADIUS R CENTERED AT RED POINTS, TO GET A (NON CONVEX) LOCALE.

CONVEXITY & STAR-SHAPEDNESS

We can ask questions like:

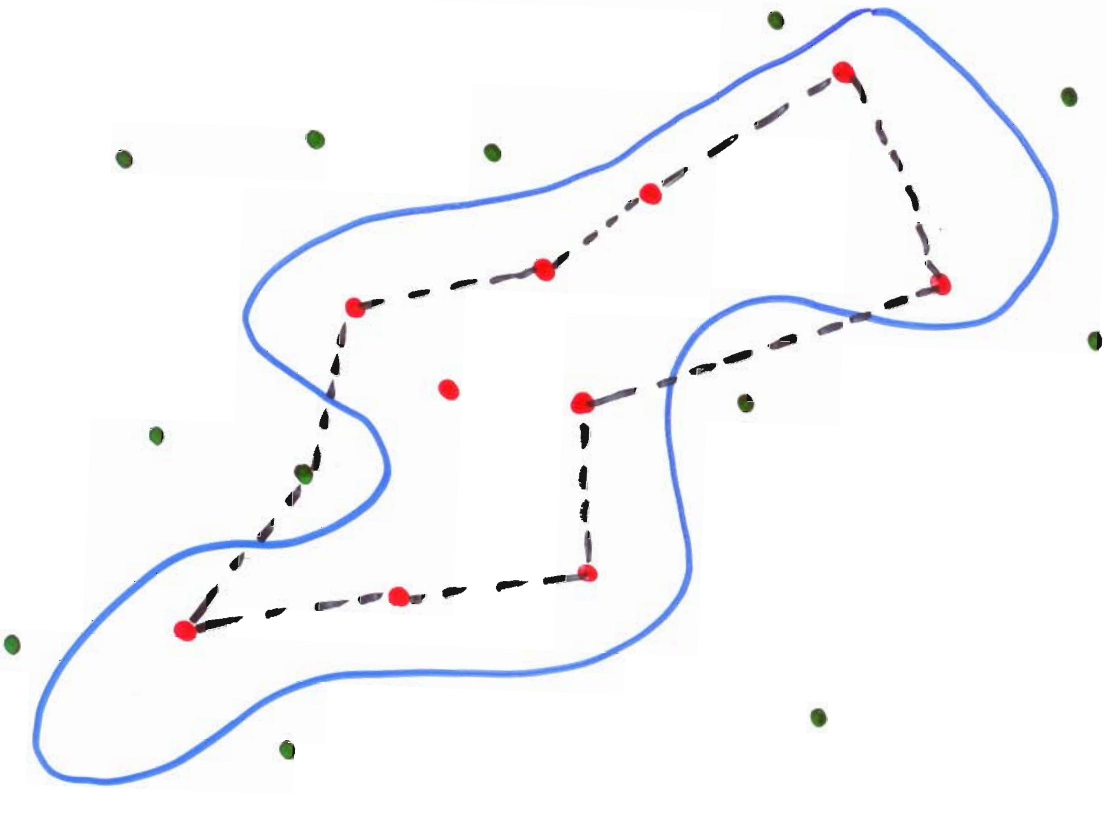
WAS THE PREIMAGE CONVEX

WAS THE PREIMAGE STAR-SHAPED

WHAT CHARACTERIZES "DIGITAL" CONVEXITY & Star-Shapedness?

In these contexts an important PREIMAGE THAT CAN BE CONSIDERED IS THE SO-CALLED

RELATIVE CONVEX HULL OR MINIMAL PERIMETER POLYGON



HOW ABOUT DESIGNING PLANAR SHAPES for GOOD METROLOGY ?

$S(x,y)$

The shape S is translated in the plane to some location (X,Y) ~ $S(x-X,y-Y)$ and digitized there to $S^D(x-X,y-Y)$.

How well can we estimate X,Y?

For given shapes we have location estimates via the "LOCALES".

- How small can locales be?
- How do we design shapes to have small locales?

How small can locales be?

- Information Theory Bound.

If the shape is of size $A \times A = A^2$ (in pixel-dimensions) the digitized image will have about A^2 bits of meaningful information. If we use 1 bit for ROUGH LOCATION TO WITHIN ONE PIXEL the rest of A^2-1 bits will be able to refine and encode the locale. So the 2^{A^2-1} bit configurations will give us the possibility to get a locale of an area not smaller than $\left(\frac{1}{2^{A^2-1}}\right)$, A LOWER BOUND ON HOW GOOD A LOCALIZATION WE CAN HAVE, in terms of Area Uncertainty

- for balanced designs the best we can hope is to get a $\Delta x, \Delta y$ precision of about $\frac{1}{2^{(A^2-1)/2}}$

AN OPTIMAL LOCATION FIDUCIAL.

Works by successive refinement

CUTE BUT NEEDS HIGH PRECISION ETCHINGS.

THE IMPORTANCE OF BEING GRAY!

Suppose we have circular pixel sensors and we do gray-scale digitization for BINARY preimages that are polygonal shapes.

WE HAVE A POSSIBILITY TO INCREASE SPATIAL REZOLUTION AT THE EXPENSE OF THE # OF SHADES OF GRAY.

Where should we invest our bits?

$$\underset{\text{TOTAL BIT USAGE}}{B} = \underbrace{N \times N}_{\substack{\text{REZOLUTION} \\ \text{in space}}} \times \underset{2^b = \#\text{gray levels.}}{b}$$

MAIN RESULT:

Once a sufficient spatial resolution is attained ~ put all the bits into gray scales!

The (Locale size) ambiguity μ

$$\boxed{\mu < \frac{1}{\sqrt{3}\,N\,(2^b-2)^2}}$$

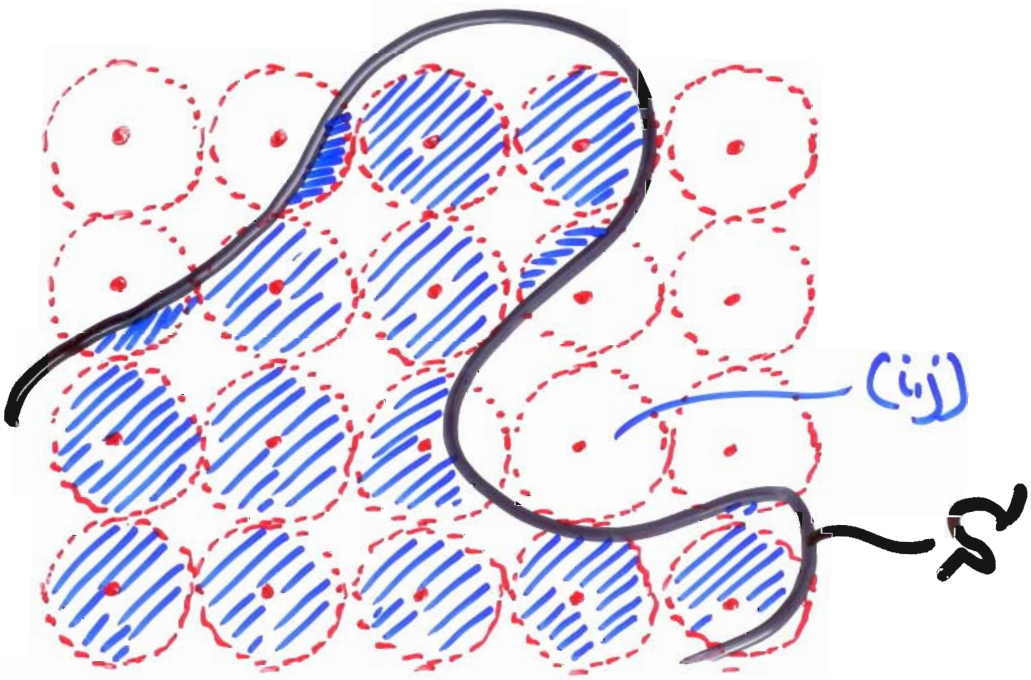
To DECREASE μ invest in $\underline{\underline{b}}$! more than in N.

$$N > f(\text{Geometry of preimages})$$

(once this is achieved put all bits into $\underline{\underline{b}}$)

EXTENSIONS

1. More realistic sampling & quantization



- GRAY LEVEL AT (i,j) $\propto$ AREA COVERED BY **S**

  subsequent quantization to 2^b-levels (b-bits/pixel).

2. Other types of pixel grids & other types of sensors too.

CHALLENGES (HOME-WORK)

- GRAY SCALE DIGITAL LINES
  - characterization ...
- Better Fiducial Designs in realistic discretization scenarios
- DIGITAL CIRCLES with B/W & Gray Scales
- Other types of Shape Probings
  - moving scans
  - irregular sensor arrays
  - tomographic probes.

CONCLUSIONS

- digital geometry is both fun & practically important
- lots of math & technological problems need still to be addressed.