Image Plane Operations

Manipulating images requires some mathematical operation to be applied to 2-D data. Some of these are very simple and straight-forward:

Translation — $f(x,y) \rightarrow f(x-a, y-b)$

Magnification — $f(x,y) \rightarrow f(ax, by)$

It is obvious what is meant by these, and some care must be taken to accommodate the finite matrix size in any actual computer implementation.

Reversal and Rotation

Both translation and magnification have one-dimensional meaning, but in 2-D we can have other kinds of operations that simply do not exist in 1-D analogs. These include some types of reversal, and also rotations.

Reversal about y-axis: $f(x,y) \rightarrow f(-x,y)$

Reversal about x-axis: $f(x,y) \rightarrow f(x,-y)$

Reversal about both: $f(x,y) \rightarrow f(-x,-y)$

Reversal about both axes is equivalent to a rotation by half a turn. The other two reversals cannot be accomplished through rotation.

In 3-D, $f(x,y,z) \rightarrow f(-x,-y,-z)$ cannot be achieved by a rotation ~~by~~ for objects not possessing symmetries.

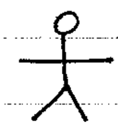
Rotation can be described by the following mapping:

$$f(x,y) \rightarrow f(\cos\theta\, x - \sin\theta\, y,\ \sin\theta\, x + \cos\theta\, y)$$

This again has no equivalence in 1-D; rotation only has a meaning in higher dimensional spaces.

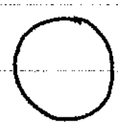
Symmetry

Symmetries also provide information about objects and have implications on their display.

Bilateral symmetry:  $f(x,y) = f(-x,y)$

Bilaterally symmetric objects are invariant to one or the other axis reversal.

Circular symmetry:

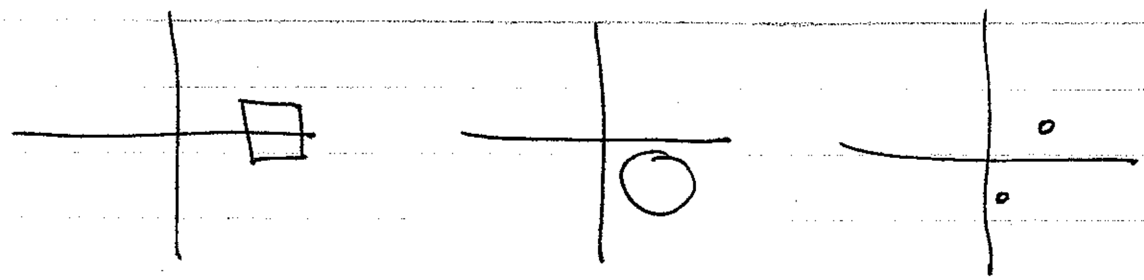
 $f(x,y) = f(x\cos\theta - y\sin\theta,\ x\sin\theta + y\cos\theta)$ for all θ

N-fold rotational symmetry:

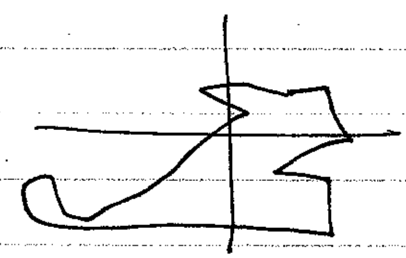
 $f(x,y) = f(x\cos\theta - y\sin\theta,\ x\sin\theta + y\cos\theta)$ for discrete, evenly-spaced values of θ $\left[\theta = \frac{2\pi}{N}\right]$

↑ 5-fold

Asymmetric objects:



However, each of these simple objects may be made symmetric by a suitable translation. This wouldn't be the case for the more complicated object below



Changing (usually reducing) Dimensionality

We have talked already about the need to reduce 3-D objects to 2-D in order to display and convey information. Let's consider here methods of extracting 1-D functions from 2-D functions.

Often we have advantages in describing 2-D data in a one dimensional sense. A simple example: in a computer, a two-d matrix is stored as a one-d array through a raster scan.

Cross-sections. Consider the common case of a straight-line cross section through the origin of a two-dimensional function. We can define a cross-section s by

$$s(x') = f(x' \cos\theta, x' \sin\theta)$$

where θ is the rotation of a primed coordinate system x', y'.

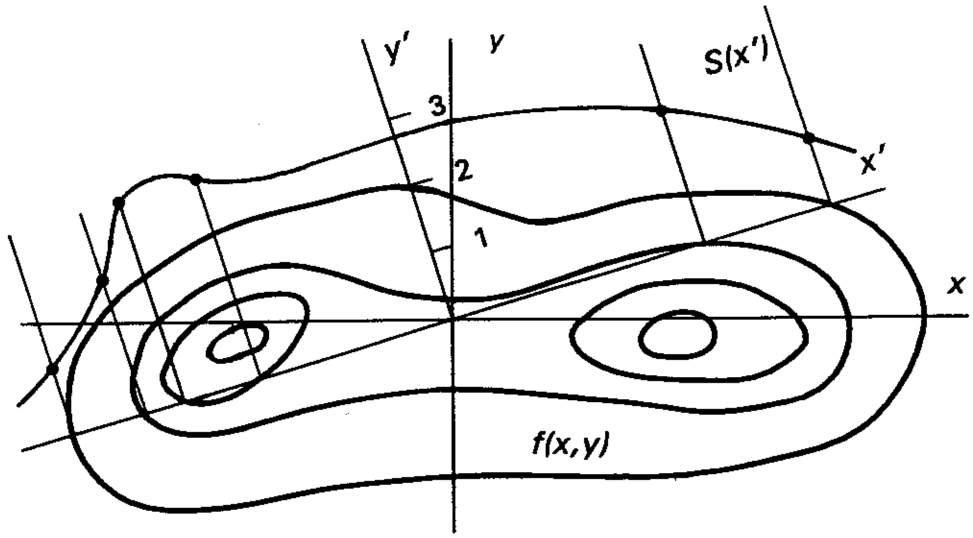


Figure 2-39 A function $f(x, y)$ and its section along the radial line $y' = 0$ or $y \cos\theta - x \sin\theta = 0$. The section $s(x')$ is shown cross-hatched and rotated through 90° into the (x, y)-plane.

This section is simply the value of the function along the rotated line $y' = 0$, or $y \cos\theta - x \sin\theta = 0$. Clearly by choosing enough cross-sections we can construct the original $f(x,y)$ completely, or perhaps a single cross section can display the needed information.

Example: height of terrain below an airplane

Raster scans

Often used in computer systems, raster scans are a convenient means to convert and image into a time series. ~~Th~~ They also form the basis of analog TV transmission.

a b c d
e f g h
i j k l
m n o p

transforms to

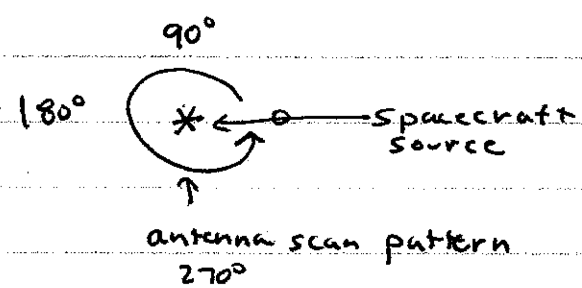
a b c d e f g h i j k l m n o p

if we adopt the convention of left-right followed by top-bottom. While natural for people raised reading European manuscripts, many other methods are possible.

With raster scan, we also need to store information of line lengths in adding to knowing the scan directions.

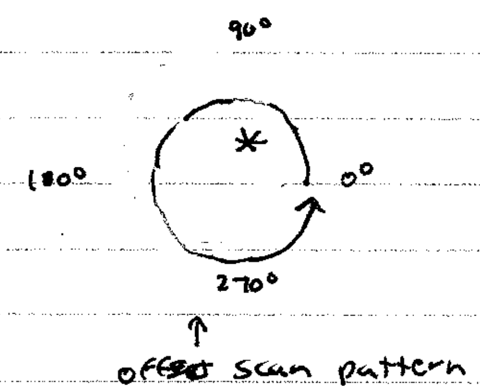
Conical scans

One very important scan geometry is conical scanning, used in tracking systems such as deep-space communications systems. Here the tracking antenna is scanned circularly and the power of the received signal analyzed to point the antenna correctly.



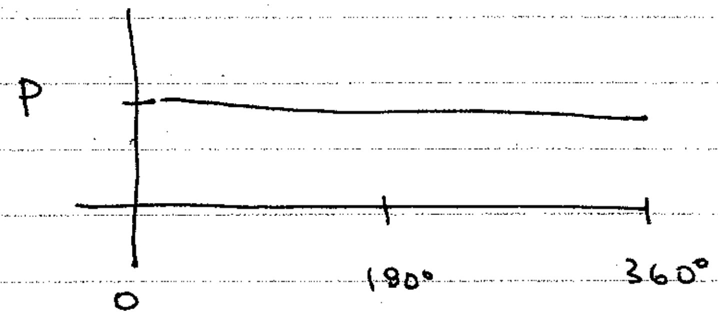
Correctly pointed

If the antenna is correctly pointed, the signal intensity will remain constant during the scan. If it is offset, the signal level will vary and the antenna can be moved in the direction of greatest signal.

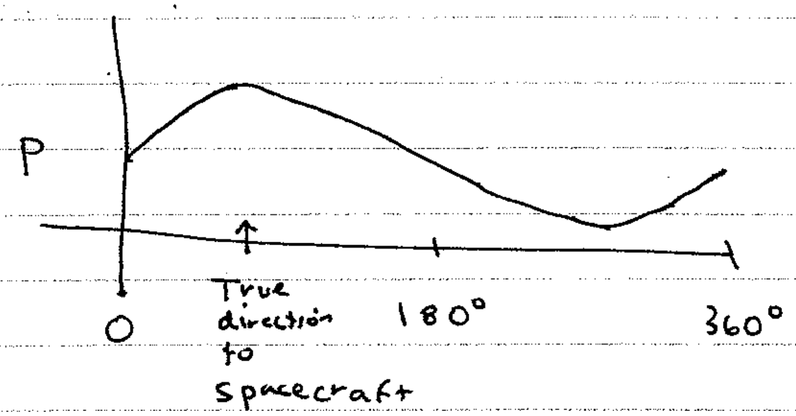


Mispointed antenna

Power vs. angle in scan, correct pointing



Power vs. ~~angle~~ angle for mispointed antenna:



<u>Stacking data</u>

The inverse of raster scanning, needed to reconstruct raster images from one-dimensional sequences, maps a 1-D function $f(x)$ into a stacked function $g(x,y)$, and is given by

$$g(x,y) = f(x - Xy) \quad , \quad 0 \le x \le X \, , \; y = 0, -Y, -2Y, \cdots$$

where X is the line length and Y is the vertical spacing.

The Voyager spacecraft carry such a 1-D sequence to be formed into raster images by aliens. We can only hope they understand stacking as well as appreciate Chuck Berry!

Projection of Images

We have mentioned projection already with regard to X-ray imaging, where we not only want to represent the outer, surface properties of objects but also want to convey information about what is inside.

Let's define a projection at an angle Θ through an object as the integral of the function along a line-of-sight perpendicular to the projection axis. For a zero angle projection, defining the projection operator $P_0 f(x,y)$ we have

$$g_0(x) = P_0 f(x,y) = \int_{-\infty}^{\infty} f(x,y)\, dy$$

← Note typo in book

The projection $g_0(x)$ is simply the sum of the function along the y direction, expressed as a function of x.

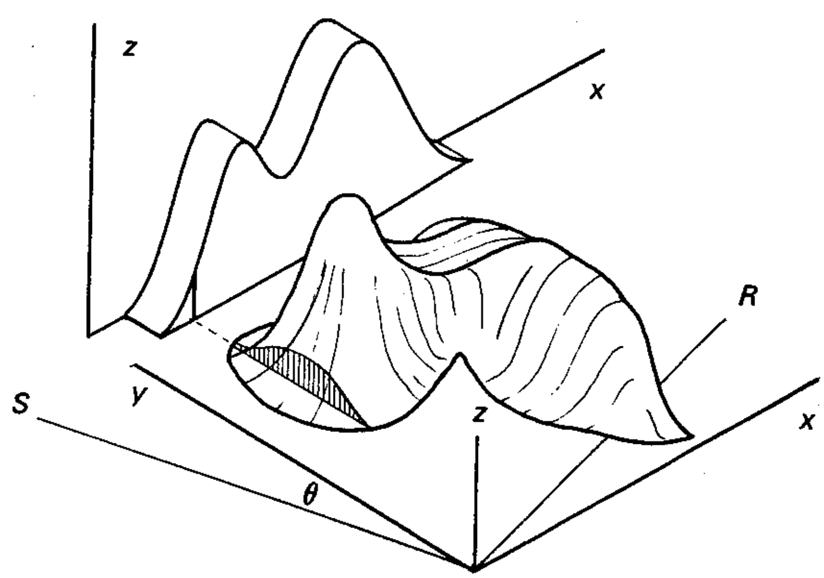


Figure 2-41 The projection operator $\mathcal{P}_\theta$. A two-dimensional function $z = f(x, y)$, represented here by a double-humped mound of clay, generates a one-dimensional function $\mathcal{P}_0 f(x, y) = \int f(x, y)\, dy$, which is a function of x when the clay is projected back at an angle $\theta = 0$ with the y-axis onto the (x, z)-plane. When the projection takes place in a direction making an angle θ with the y-axis, the result is $\mathcal{P}_\theta f(x, y) = \int f(x, y)\, dS$, which is a function of the rotated coordinate R.

We can generalize the projection for arbitrary angle θ:

$$g_\theta(R) = P_\theta f(x,y) = \int_{-\infty}^{\infty} f(x,y)\, ds$$

where R and S are rotated coordinates resulting from a counterclockwise rotation of (x,y) through θ.

For a function of constant density, such as a clay model, the numerical value of a projection at a given point is simply the area of the plane cross-section at that point.

If the function is circularly symmetric, all projections would be equal. For example, let $f(x,y) = \exp(-\pi r^2)$

$$g(x) = \int_{-\infty}^{\infty} e^{-\pi(x^2+y^2)}\, dy = e^{-\pi x^2}\int_{-\infty}^{\infty} e^{-\pi y^2}\, dy = e^{-\pi x^2}$$

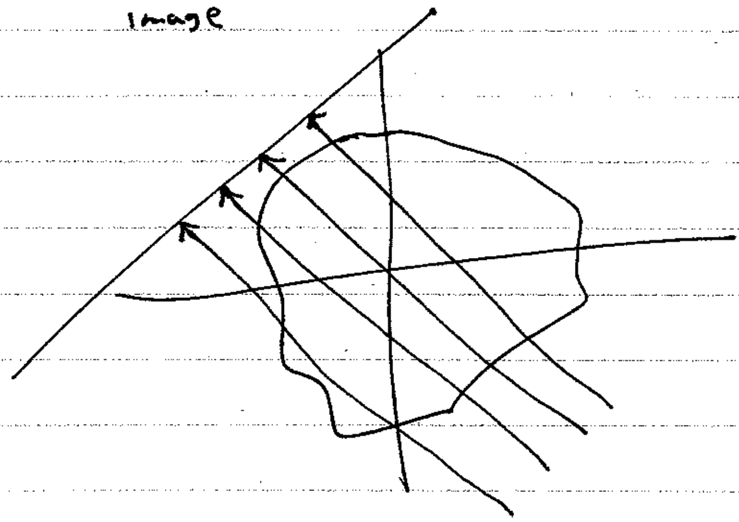
For the circular pillbox $f(x,y) = \text{rect}(r)$

$$g(x) = \sqrt{1-4x^2}\ \text{rect}\ x$$

A projection in tomography is called a <u>scan</u>. In each case it is determined by a line integral along a scanning beam.

<u>Dimensions</u>

The projection operator we have presented maps a two-dimensional function into a one-dimensional scan, even though we have used a three-dimensional model to understand the operation. What we are really doing is calculating the line integral along a plane of a two-d image



The image is assumed to have a varying value, that is it has "shape". Shape is simply a means for us to visualize that we have a function that depends on location in the plane.

3-D Projections

3-D functions project onto a plane in much the same way as 2D functions project onto a line. Again the projection represents a series of line integrals through the object. For example

$$g_{\substack{\text{onto}\\ x\text{-}y\\ \text{plane}}}(x,y) = \int_{-\infty}^{\infty} f(x,y,z)\,dz$$

Each spot on the ~~sk~~ projection plane corresponds to an integral as did each point on the line of a 2-D projection.

Silhouetting

A silhouette is a special projection of a 3-D image, where the function is either a 0 or 1 depending on whether the illuminating ray encounters the object. It is used most commonly in graphic art contexts, but it does find use in some scientific analyses.

Back projection

Inversion of projection is called back projection -- it is filling a 2-D array from a 1-D projection. Clearly the projection operation is not reversible, yet back projection respresents one step toward reconstructing a full image from 1-D data.

In its most simple form, a plane is filled ~~on~~ with a density equal to the value of the projection, with the resulting distribution inclined at the proper projection angle:

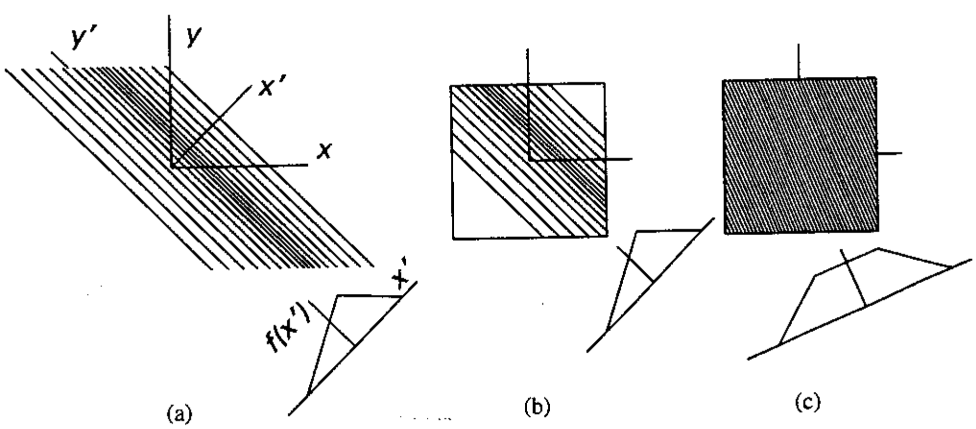


Figure 2-42 (a) Back projection, (b) truncated back projection, (c) conservative back projection.

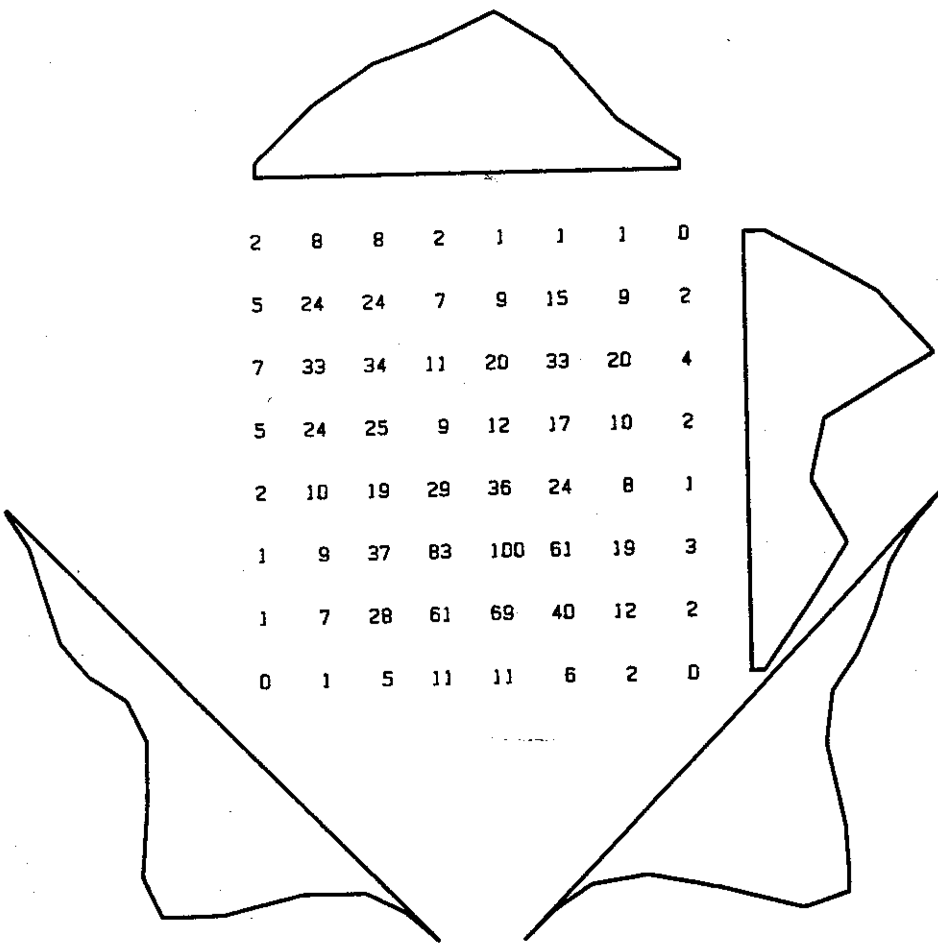
The mapping quite simple:

$$B\, f(x') = f(x')$$

one-d function — entire 2-D plane parameterized by x'

The back projection may be truncated in the image plane if we know that it is of limited extent.

Projection of a digital image



2	8	8	2	1	1	1	0
5	24	24	7	9	15	9	2
7	33	34	11	20	33	20	4
5	24	25	9	12	17	10	2
2	10	19	29	36	24	8	1
1	9	37	83	100	61	19	3
1	7	28	61	69	40	12	2
0	1	5	11	11	6	2	0

Figure 2-47 A 64-element map and four equispaced projections.