Spatial Transformations

Note Title 5/9/2006

Consider $I(x,y)$, an image.

Say you generate a new image

$$I'(x',y') = I(x,y)$$

where $x' = r(x,y)$
$y' = g(x,y)$

Then I' is a spatial transformation of I.

Eg. $x' = \frac{x}{2}$

$y' = y/2$

$\therefore I'\left(\frac{x}{2}, \frac{y}{2}\right) = I(x,y)$

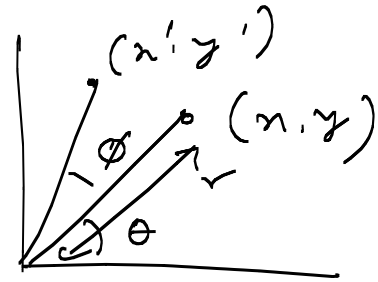
$\therefore$ Scaled down by 2, I' is a subsampled version of I.

In general,

$$x' = s_x x$$
$$y' = s_y y.$$

$$\therefore \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

<u>Rotation</u>



$$x = r\cos\theta$$
$$y = r\sin\theta$$

$$\therefore x' = r\cos(\theta+\phi)$$
$$y' = r\sin(\theta+\phi)$$

$$x' = r\cos\theta\cos\phi + r\sin\theta\sin\phi$$
$$y' = r\sin\theta\cos\phi - r\sin\phi\cos\theta$$

$$x' = x\cos\phi + y\sin\phi$$
$$y' = y\cos\phi - x\sin\phi$$

$$\therefore \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

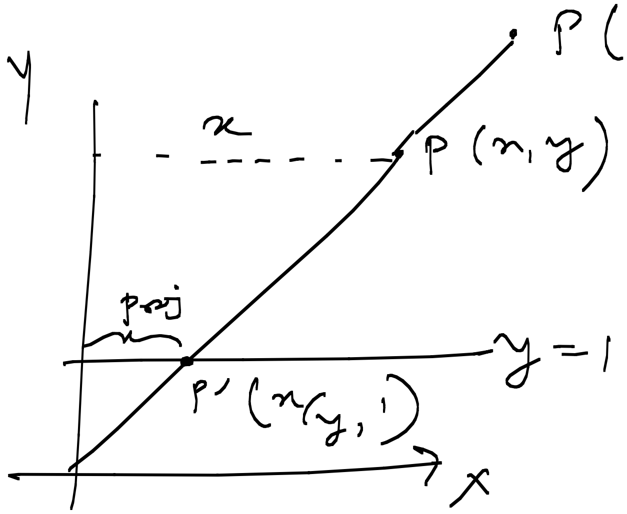
Translation

$x' = x + t_x$

$y' = y + t_y$

Now how do we represent in matrix?

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



3D Homogeneous Coordinate

Find projection of P on y = 1

$$\frac{x}{proj} = \frac{y}{1} \quad \therefore proj = \frac{x}{y}$$

$\therefore$ A pt. on an 1D line can be seen as the projection of 2D pt. $\therefore$ What can be expressed by 1D coordinate, can be represented by a 2D Homogeneous Coordinate

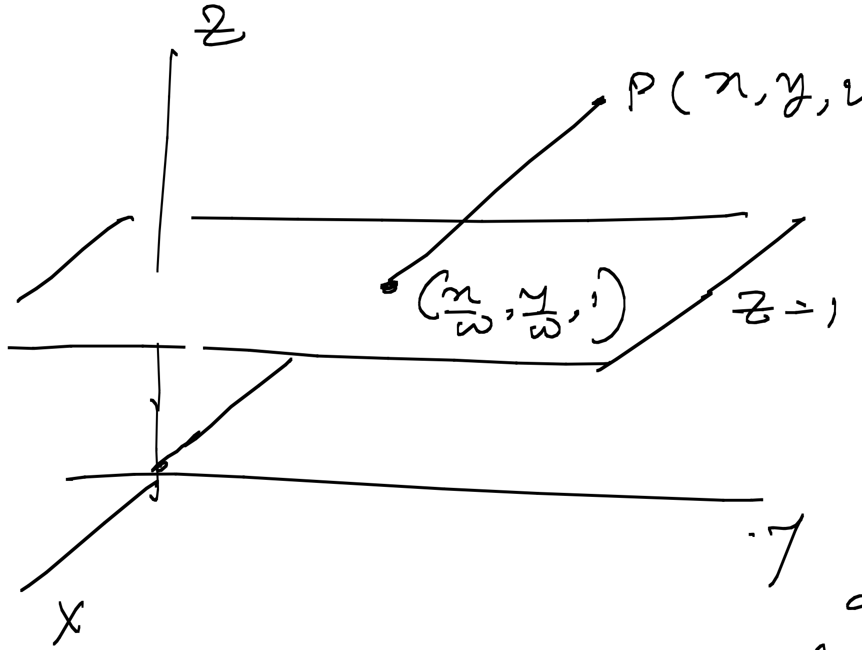
Consider another pt. on the same line, $(2x, 2y)$

Has the same projection, $\frac{x}{y}$

Similarly $\left(\frac{2x}{y}, 2\right)$, $\left(\frac{3x}{y}, 3\right)$ are all on same ray & have same coordinate on the 1D line.

$$(x, w) \longrightarrow \left(\frac{x}{w}, 1\right)$$

Now extend this idea to 2D



∴ 2D pt. on a plane can be thought of a 3D homogenous coordinate pt.

Extend the same idea to 3D.

$$\therefore (x, y, z, w) \longrightarrow \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}, 1\right)$$

$$\therefore \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Now express (x', y') also in homogeneous coordinate

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scaling

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Rotation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Shear

x-shear

$y' = y + ax$

$x' = x$

$$\therefore \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

y-shear

$$x' = by + x \qquad \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & b & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$y' = y$$

Concatenation of Transformation

S R T

= Just multiply the matrices to generate the transformation.

Will generate one 3X3 matrix

∴ Lets assume that I under goes a unknown transformation T which can be a series of linear transformation also.

Linear means,

Say you have a st line

$aP + bQ$ where P & Q are the end pts of the st. line

$$T(aP + bQ) = T(aP) + T(bQ)$$

$= a\,T(P) + b\,T(Q)$

$\therefore$ Transforming vertices is sufficient. Do not need to transform all pts on the line. Makes transformations easier.

Transform end pts, use linear interpolation to generate intermediate pts.

Finding unknown transformation given I & I'.

say $I' = T(I)$

Now we know $\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix}$

Note T has 9 unknowns, $T(3,3) = 1$. $\therefore$ 8 unknowns.

If we can detect pts in I which correspond to I'

Correspond is a very strong word.
Think of it to be done even manually

$$\therefore \begin{bmatrix} x'w \\ y'w \\ w \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\therefore wx' = a_1 x + a_2 y + a_3 \quad — (1)$$

$$wy' = a_4 x + a_5 y + a_6 \quad — (2)$$

$$w = a_7 x + a_8 y + 1 \quad — (3)$$

$$\therefore x' = \frac{a_1 x + a_2 y + a_3}{a_7 x + a_8 y + 1}$$

$$\sim a_7 x' x + a_8 y x' + x' - a_1 x - a_2 y - a_3 = 0 \quad — (4)$$

Similarly from (2) & (3)

$$a_7 x y' + a_8 y y' + y' - a_4 x - a_5 y - a_6 = 0 \quad — (5)$$

Now if we find one correspondance,
we know $(x_1', y_1') \longleftrightarrow (x, y)$
Plug it in (4) & (5) [Remember a_is are unknown]

we get 2 eqn in term of a_i's.
We have 8 unknowns ∴ need
8 eqn to solve for a_i's.
Each correspondence generates 2 eqn
∴ Need 4 eqn to find the
a_i's & hence T.

Now say if T involves only
rotation & Scaling. ∴ No need
of homogeneous coordinates.

$$\therefore \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

∴ Four unknowns.
∴ only 2 correspondence would
suffice.

Non linear transformation

$$x' = a_0x^2 + a_1y^2 + a_2xy + a_3x + a_4y + a_5$$

$$y' = b_0x^2 + b_1y^2 + b_2xy + b_3x + b_4y + b_5$$

$\therefore$ 12 unknown parameters.

If we known correspondence, we get two eqn from each correspondence. $\therefore$ Need 6 pts to find parameters for Quadratic fn.

As the degree of the fn increases, no. of unknowns increase, $\therefore$ No. of required correspondences increase

Eg. Cubic, 20 parameter. $\therefore$ 10 correspondence.

Note that using exact no. of correspondence reqd to solve the system of linear eqn increases the dependency on the correspondence.

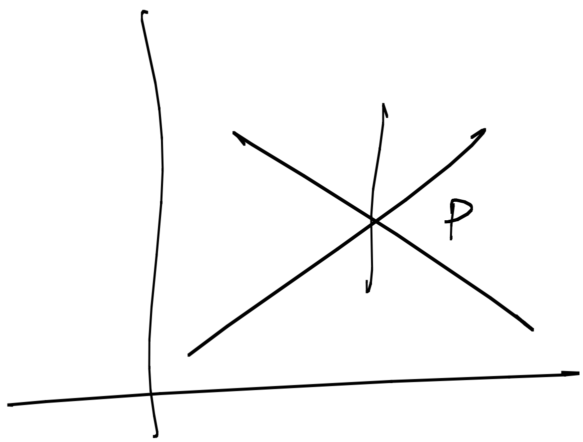
∴ Correspondence needs to be very accurate. Even if you click manually, there may be error. If it is automated feature detection, accuracy may not be very high for some pts.

∴ Usually a large no. of correspondences are found.

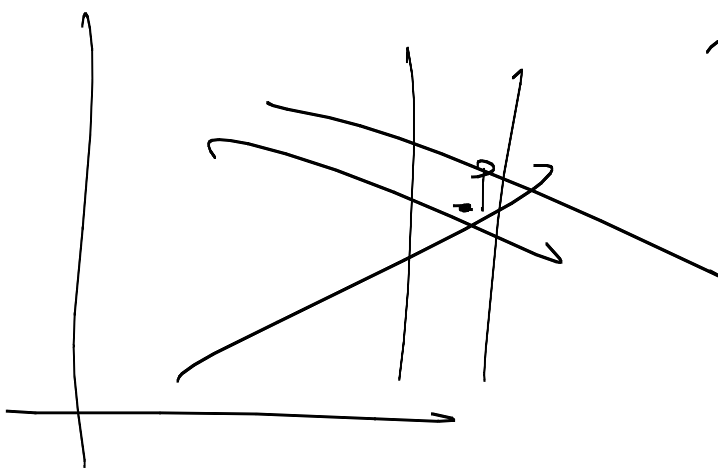
Say Quadratic Eqn,
need only 6 correspondences
But use a 10 x 10 grid, ∴ 100 correspondences
∴ Create an over-constrained system (# of Eq > # of unknowns)
Solve using least square.

When solving a set of eqⁿ find the pt. of intersection. (Think in 2D)

∴ solving for 2 variables.



But when more no. of eqⁿ



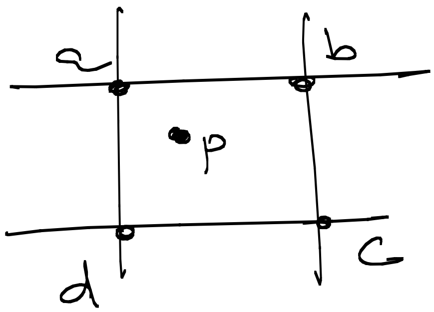
find a P s.t. sum of distances from all the lines are minimized

Same concept extended to higher dimension when solving for larger # of unknowns.

Now say you are generating I′ from I. ∴ You know the fⁿ

Say $x' \to$ floating pt. (All are FLOPS)

$\therefore$ Lands on a pixel in I $\neq$ integer



How do you generate the color?

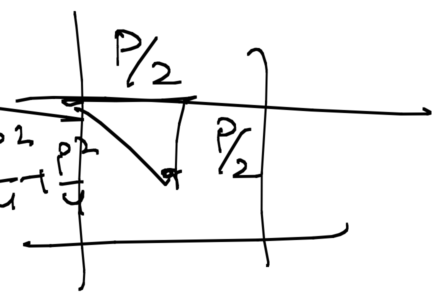
Interpolated from the neighborhood.

a) Nearest neighbor

— Eq. F(a)

Results in a spatial offset error

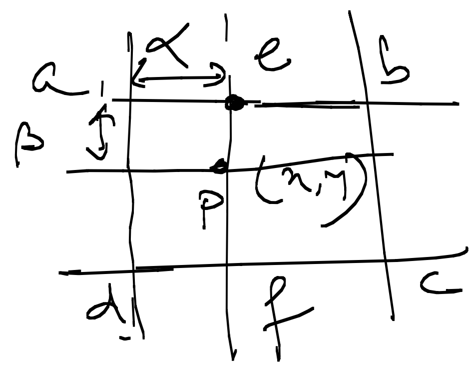
Max.



$$\sqrt{\frac{p^2}{4}+\frac{p^2}{4}} = \frac{p}{\sqrt{2}}$$

$\therefore$ Max error $= \dfrac{p}{\sqrt{2}}$

b) Bilinear interpolation



$\alpha = x - a$

$\beta = y - b$

$f(e) = \alpha F(b) + (1-\alpha) F(a)$

$F(f) = \alpha F(c) + (1-\alpha) . F(d)$

$F(P) = \beta . F(f) + (1-\beta) F(e)$

First interpolate in x direction & then in y direction.

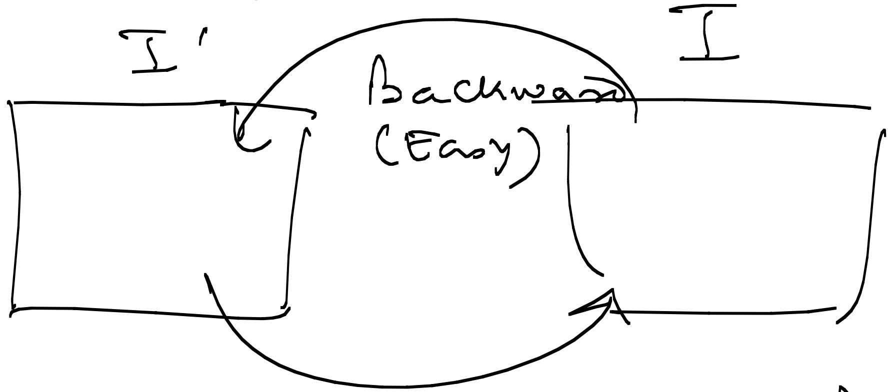
$= \beta . (\alpha F(c) + (1-\alpha) F(d)) + (1-\beta)(\alpha F(b) + (1-\alpha) F(a))$

$= \alpha\beta F(c) + (1-\alpha)\beta . F(d) + (1-\beta)\alpha F(b) + (1-\alpha)(1-\beta) F(a)$

Note though linear in each direction, results in a nonlinear surface fit between 4 neighbors.

$\therefore$ Non linear eqn of α & β.

How to generate $I'(x', y')$ from I?



Way I

$x' = g(x, y)$, $y' = h(x, y)$

Walk on the pixels in I'.

∴ for $x' = 1$ to width_I'
for $y' = 1$ to height_I'
Find (x, y) in I which
(x', y') maps to.
Interpolate in I.
end.

Note we need to know g^{-1}, and h^{-1}, which is kind of difficult, may be impossible at times.

Way II

Instead walk along the pixels in

I. $I'(x', y')$ is black

For $x = 1$ to width_I

for $y = 1$ to height_I

Find (x', y') in I';

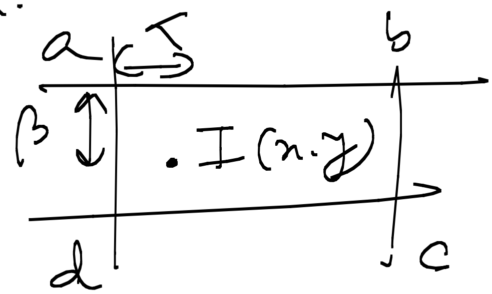
(may not be integer)

Distributed value to the neighborhood and add to existing value;

end

end.

∴



$$\therefore\ I'(a) = (1-\alpha)(1-\beta)\, I(x,y) = h_a I(x,y)$$

$$I'(b) = (1-\beta)\,\alpha\, I(x,y) = h_b I(x,y)$$

$$I'(c) = \alpha\beta\, I(x,y) = h_c I(x,y)$$

$$I'(d) = (1-\alpha)\,\beta\, I(x,y) = h_d I(x,y)$$

Note, $h_a + h_b + h_c + h_d = 1$

∴ $I(x,y)$ is distributed completely to neighborhood.