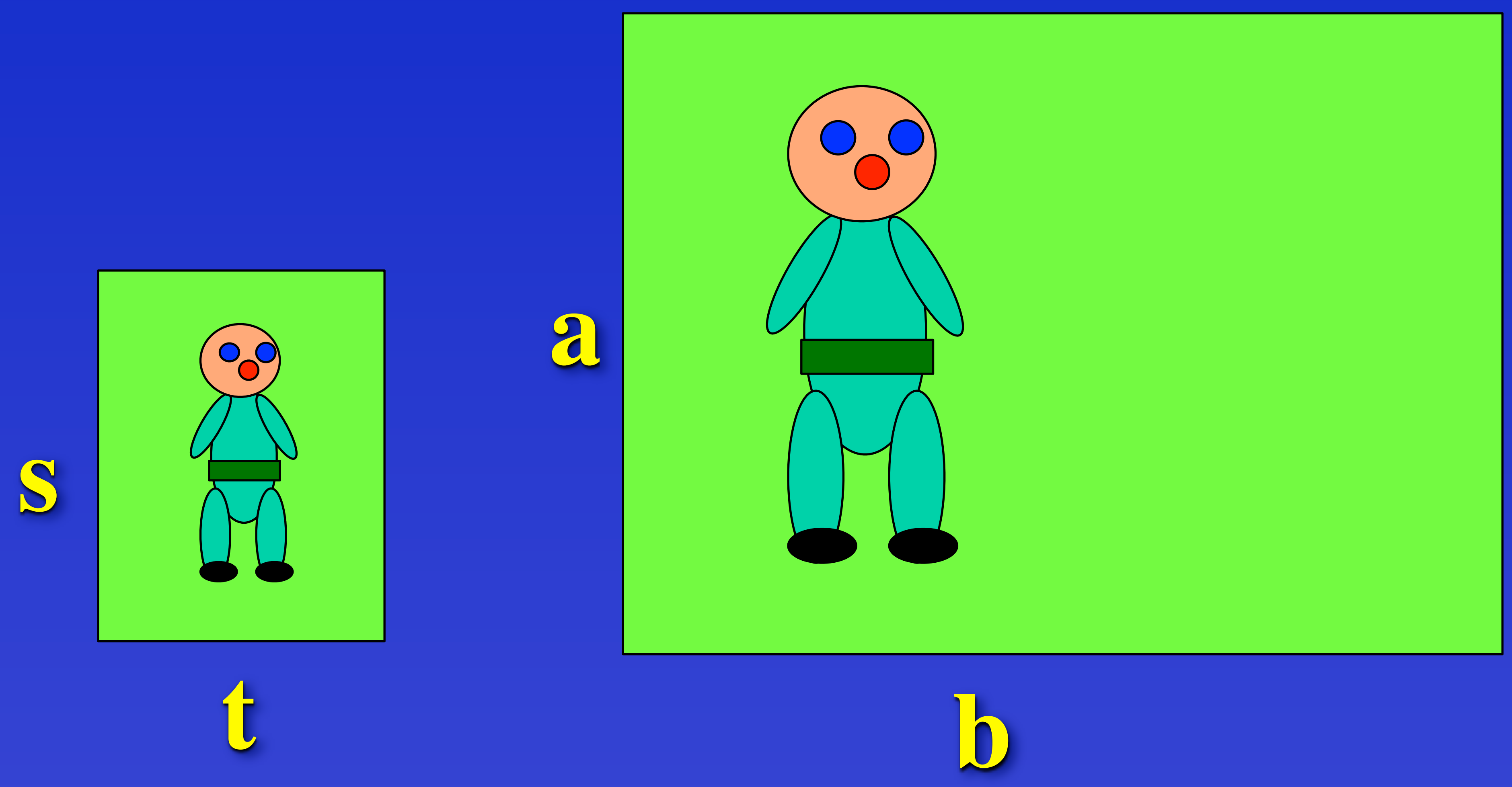
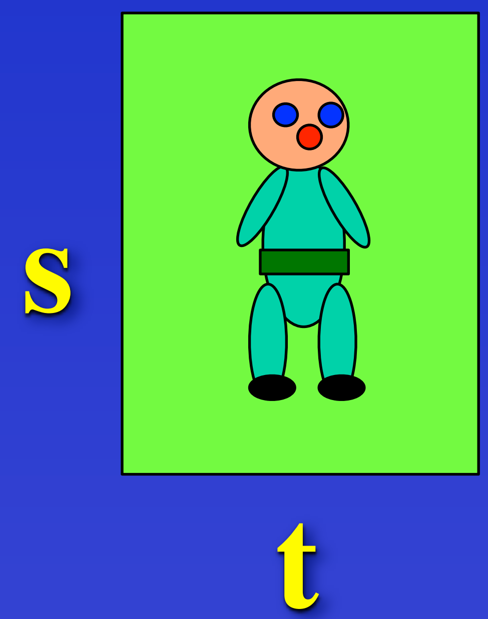
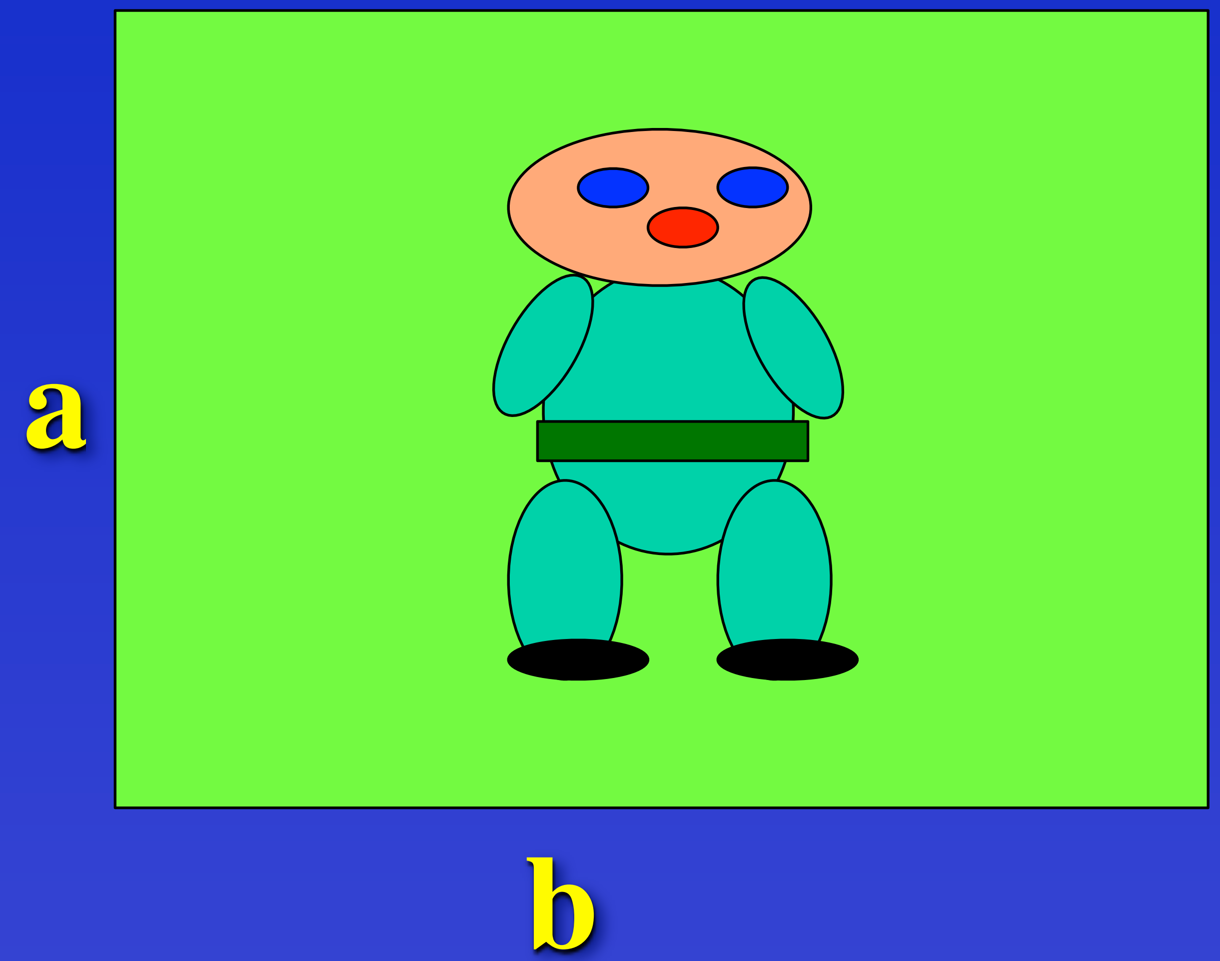
Typical scaling



Typical Blunder





OUSA Otago University Students Association
Nga Akonga o te Whare Wananga o Otakou

DO YOU WANT TO…

- Gain valuable skills in communication and negotiation?
- Help to resolve concerns students have with the lecturer or department?
- Act as a contact point between the class and the OUSA, so major concerns are dealt with by trained advocates?
- Get a certificate for your efforts?

WE NEED YOU

TO BE A CLASS REP

VOLUNTEER NOW!

Any queries email: education@ousa.org.nz or
educ.off@ousa.org.nz
Phone: 479-5449

OUSA

Otago University Students Association
Nga Akonga o te Whare Wananga o Otakou

DO YOU WANT TO…

- Gain valuable skills in communication and negotiation?
- Help to resolve concerns students have with the lecturer or department?
- Act as a contact point between the class and the OUSA, so major concerns are dealt with by trained advocates?
- Get a certificate for your efforts?

WE NEED YOU

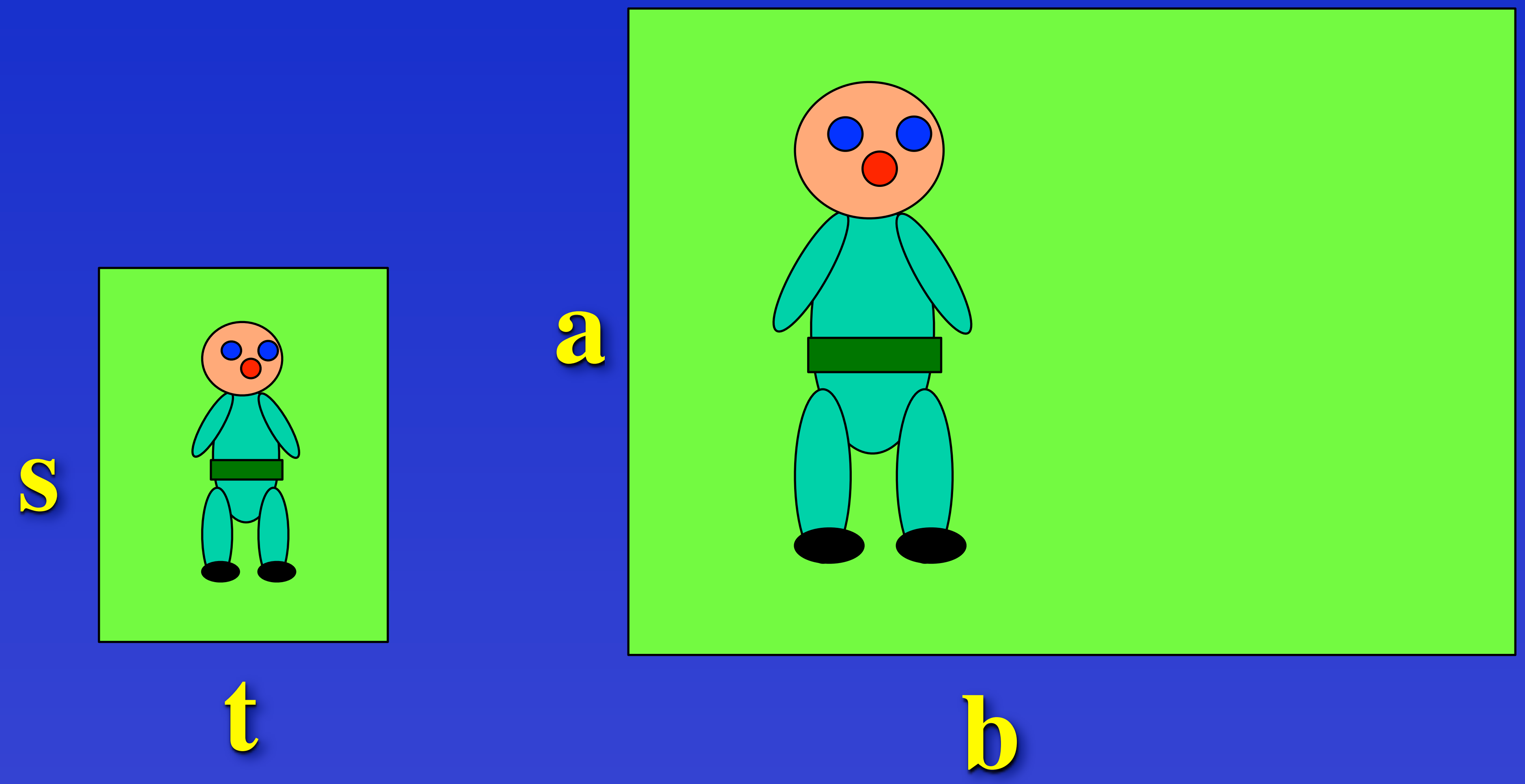
TO BE A CLASS REP

VOLUNTEER NOW!

Any queries email: education@ousa.org.nz **or** educ.off@ousa.org.nz

Phone: 479-5449

Typical scaling

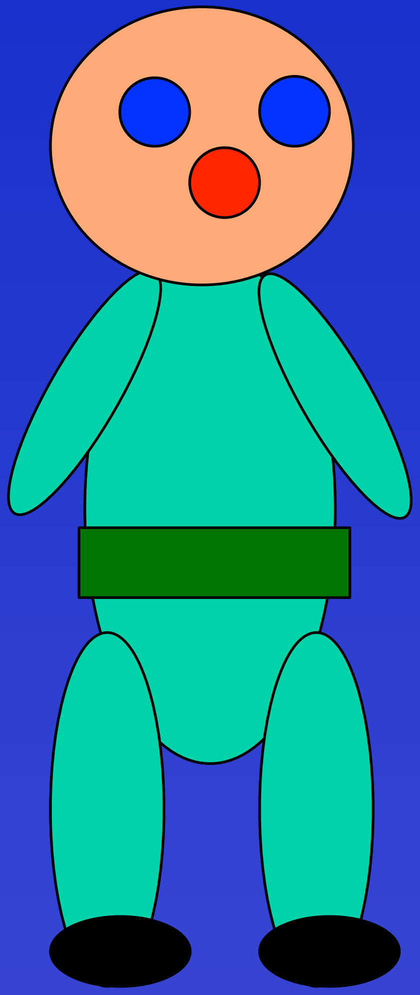


Typical scaling algorithm

```
if (a/s > b/t)
    scale = b/t
else
    scale = a/s

for (all x, y)
    x = scale * x
    y = scale * y
```

Other operations



Other operations

- **Stretch**

  $x' = x * h;$

  $y' = y * v;$

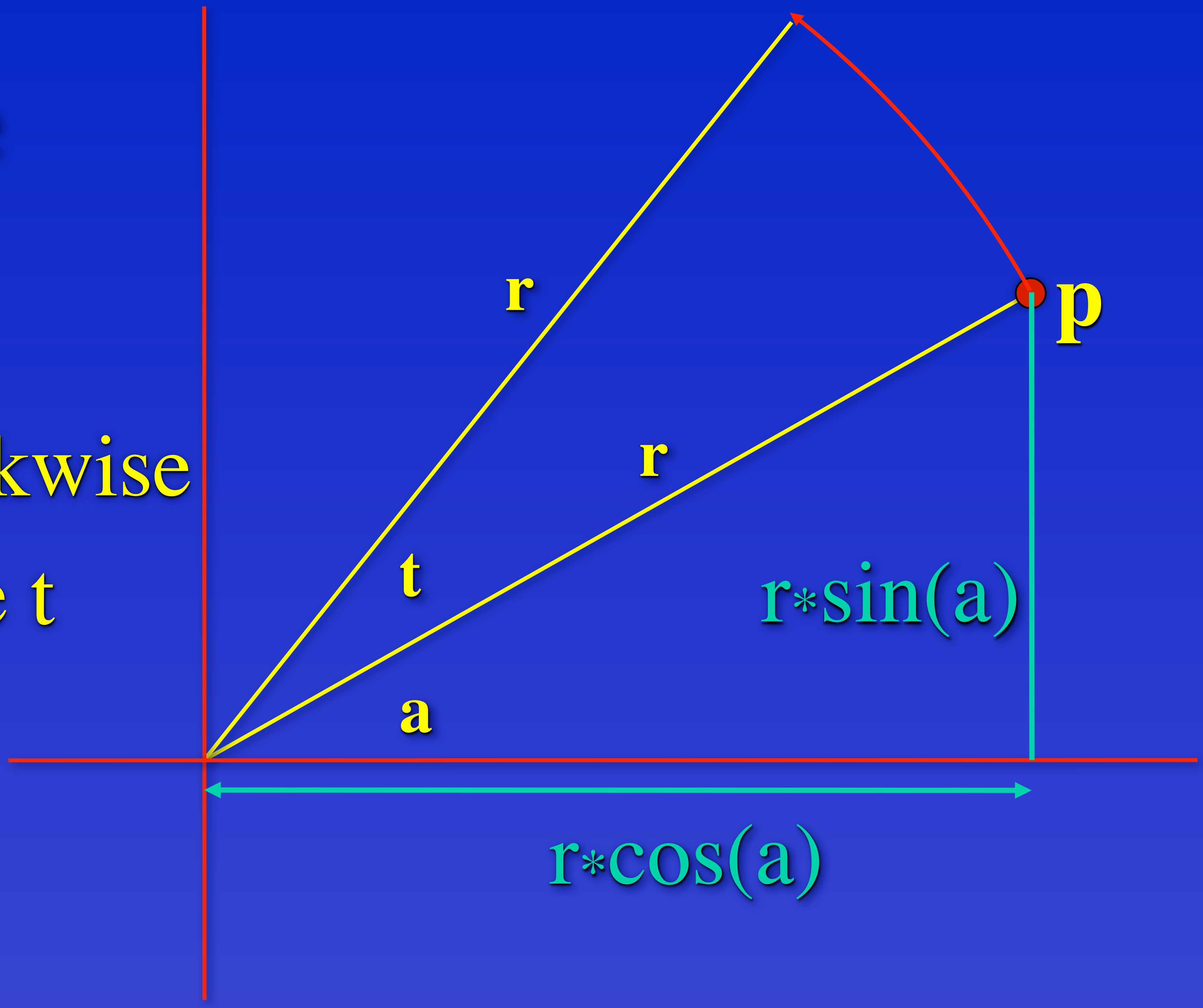
- **Shift**

  $x' = x + a;$

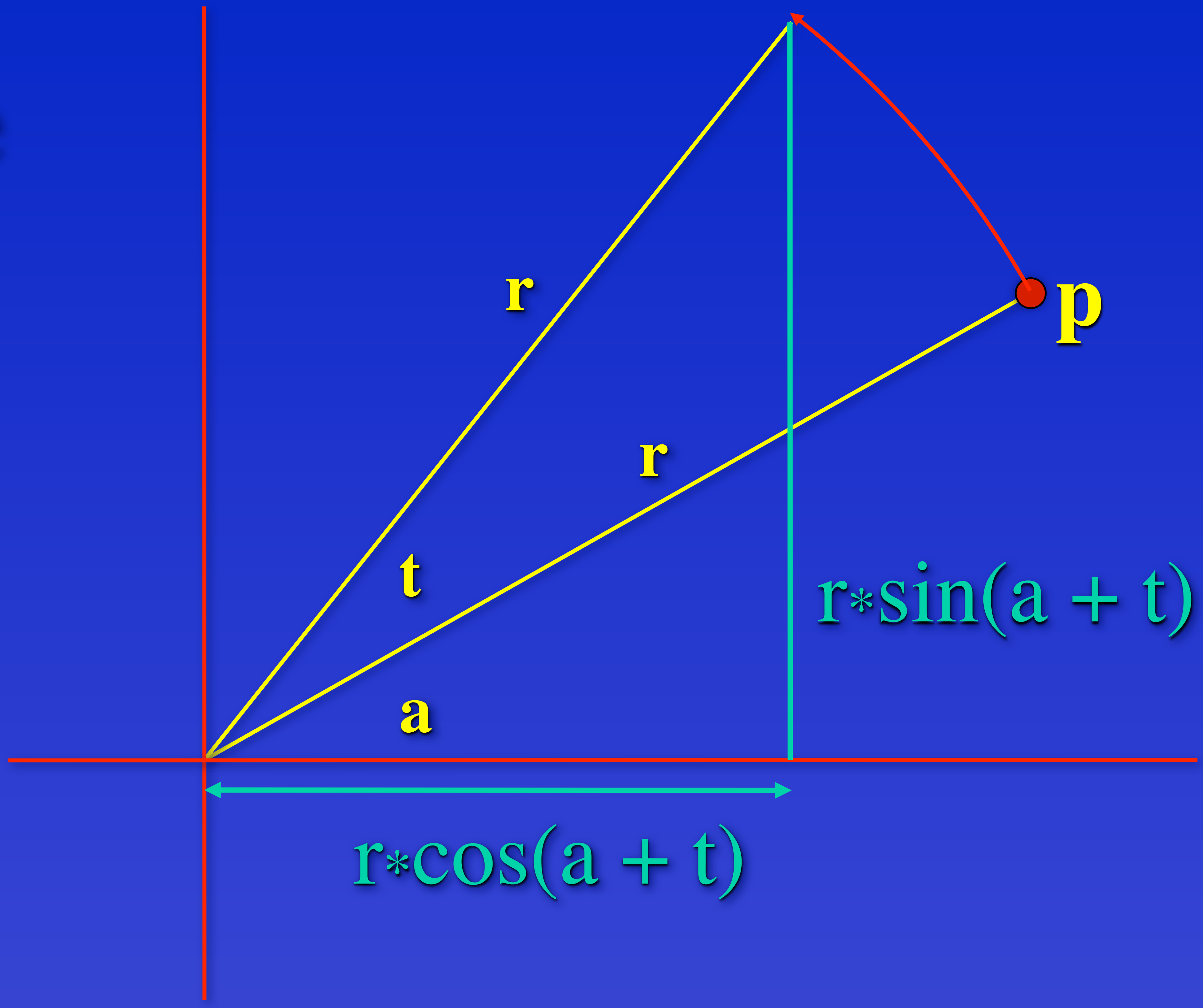
  $y' = y + b;$

Rotate

- Point **p**
- anticlockwise by angle t



Rotate



Expanding those a+t angles...

$x' = r*\cos(a + t)$

$x' = r*(\cos(a)*\cos(t) - \sin(a)*\sin(t))$

$x' = r*\cos(a)*\cos(t) - r*\sin(a)*\sin(t)$

$x' = r*\cos(a)*\cos(t) - r*\sin(a)*\sin(t))$

$x' = x*\cos(t) - y*\sin(t)$

Expanding those a+t angles...

$$y' = r * \sin(a + t)$$

$$y' = r * (\cos(a) * \sin(t) + \sin(a) * \cos(t))$$

$$y' = r * \cos(a) * \sin(t) + r * \sin(a) * \cos(t)$$

$$y' = r * \cos(a) * \sin(t) + r * \sin(a) * \cos(t)$$

$$y' = x * \sin(t) + y * \cos(t)$$

$$x' = x*\cos(t) - y*\sin(t)$$

$$y' = x*\sin(t) + y*\cos(t)$$

Notice that t is constant

```
si = sin(t)
co = cos(t)
for (all points p) do
    tmp = p.x
    p.x = co * p.x - si * p.y
    p.y = si * tmp + co * p.y
```

General object transformation

- All of our computer graphics objects consist of points or ways to find points.
- Any rotation, magnification or shift (translation) can be applied point by point.

Matrix form: scale

- The magnification by m:

$x' = m * x$

$y' = m * y$ can be written:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrix form: rotate

- Rotation counter-clockwise by angle t:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos t & -\sin t \\ \sin t & \cos t \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

But what about shift?

$$x' = x + a$$
$$y' = y + b$$

Why write (x, y) as $\begin{bmatrix} x \\ y \end{bmatrix}$?

We can use a new form:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \text{ means } (x, y)$$

$$\begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + a \\ y + b \\ 1 \end{bmatrix} \text{ means } (x + a, y + b)$$

Rotation in our new form:

$$\begin{bmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Is this matrix stuff any use?

- Very much so!
- Matrix multiplication is associative

$$(A\ B)\ C = A\ (B\ C)$$

So ...

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 7 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix}$$

Shift x by 1, y by 2

Rotate 90°

$$\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix} = \begin{bmatrix} -9 \\ 4 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -9 \\ 4 \\ 1 \end{bmatrix} = \begin{bmatrix} -18 \\ 8 \\ 1 \end{bmatrix}$$

Magnify 2x

This can be expressed:

M (R (S u))

= (M R S) u

Determine the transformation matrix

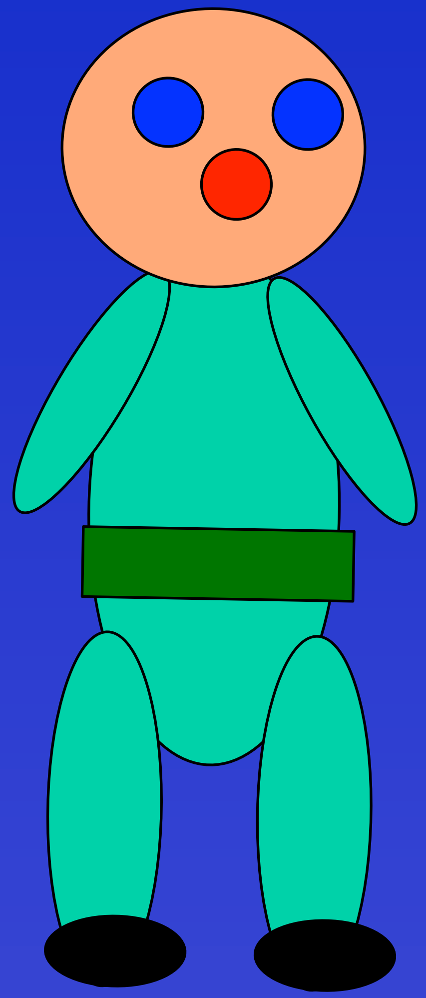
$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -2 & -4 \\ 2 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix}$$

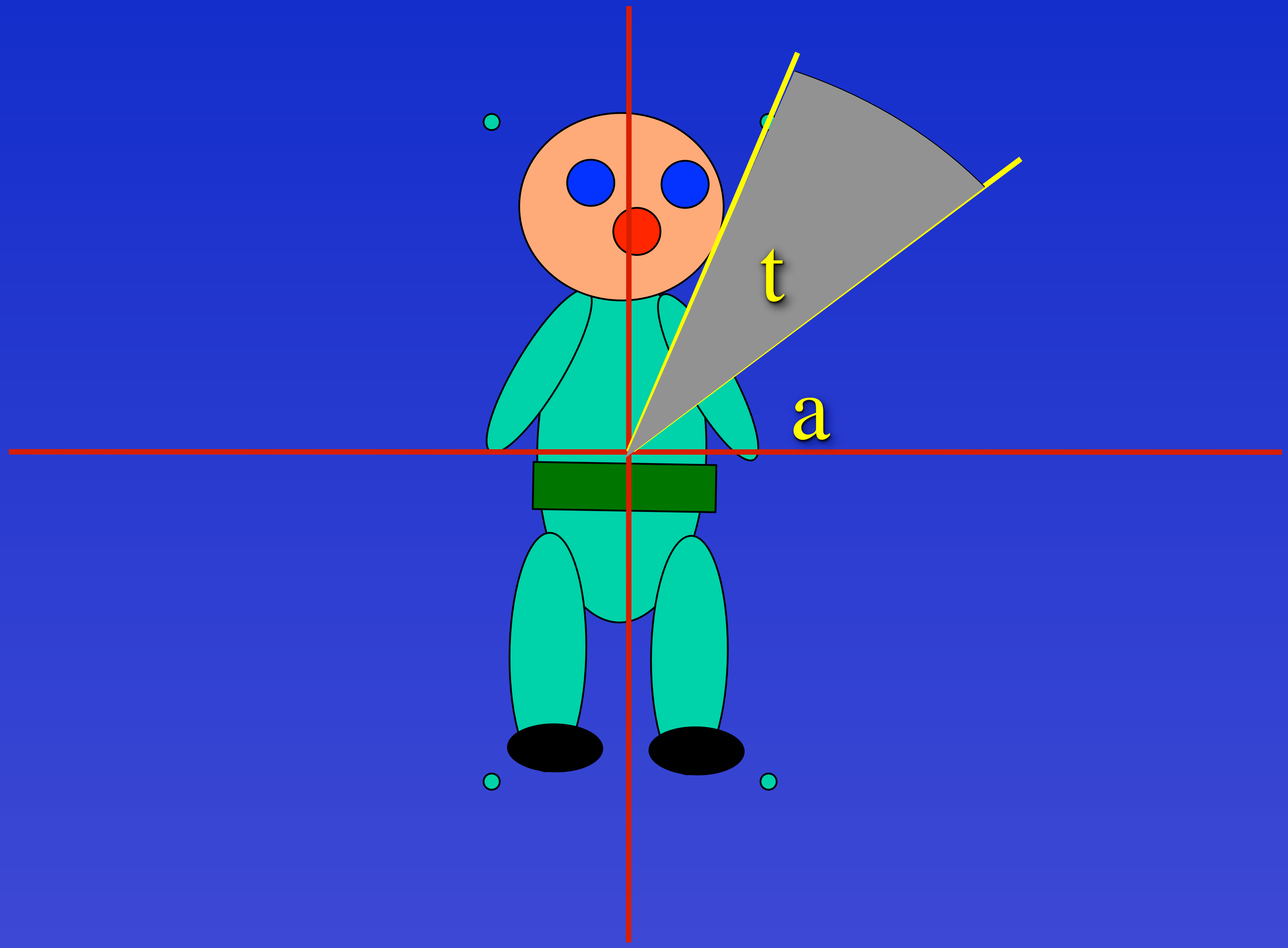
Computation savings!

- Say we have 50 operations and 1,000,000 points to transform.
- We do 50 matrix multiplications and then apply the result 1,000,000 times.
- So that is 1,000,050 operations instead of 50,000,000!

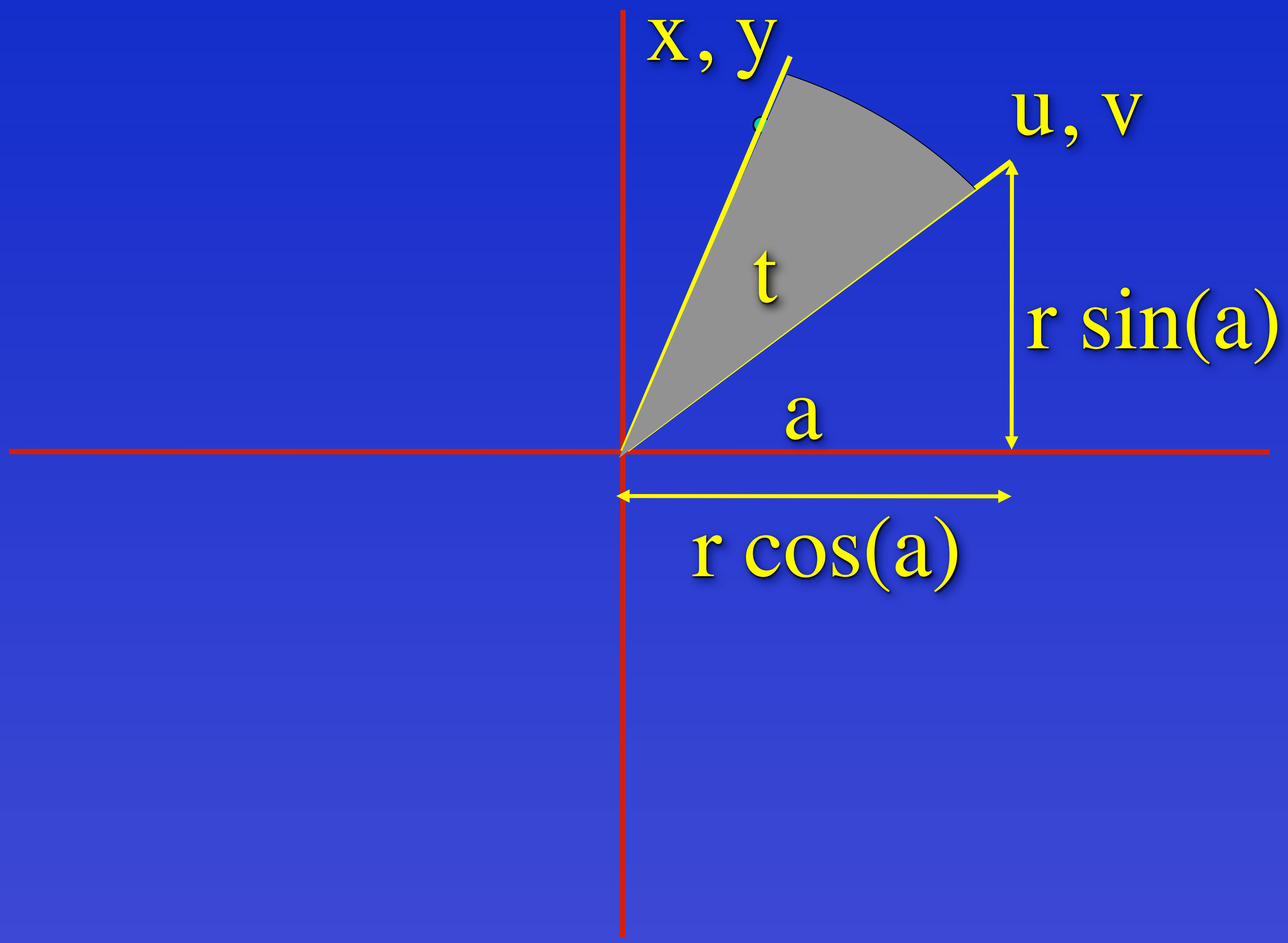
More on rotation



How do we know the angle t?



Mouse dragged: (u,v) to (x,y)



Difference of angles ...

After dividing by the radius $\sqrt{x^2 + y^2}$ or $\sqrt{u^2 + v^2}$

we have $x = \cos(a + t)$, $y = \sin(a + t)$,

$u = \cos(a)$, $v = \sin(a)$.

$\cos(a + t) = \cos(a)\cos(t) - \sin(a)\sin(t)$

$x = u\cos(t) - v\sin(t)$

$\sin(a + t) = \sin(a)\cos(t) + \cos(a)\sin(t)$

$y = v\cos(t) + u\sin(t)$

Difference of angles ...

After dividing by the radius $\sqrt{x^2 + y^2}$ or $\sqrt{u^2 + v^2}$

we have $x = \cos(a + t)$, $y = \sin(a + t)$,

$u = \cos(a)$, $v = \sin(a)$.

$\cos(a + t) = \cos(a)\cos(t) - \sin(a)\sin(t)$

$x = u \cos(t) - v \sin(t)$

$\sin(a + t) = \sin(a)\cos(t) + \cos(a)\sin(t)$

$y = v \cos(t) + u \sin(t)$

Equations in sin(t), cos(t)

$x = u \cos(t) - v \sin(t)$

$y = v \cos(t) + u \sin(t)$

$xv = uv \cos(t) - v^2 \sin(t)$

$yu = uv \cos(t) + u^2 \sin(t)$

$(yu - xv) = \sin(t) (u^2 + v^2)$

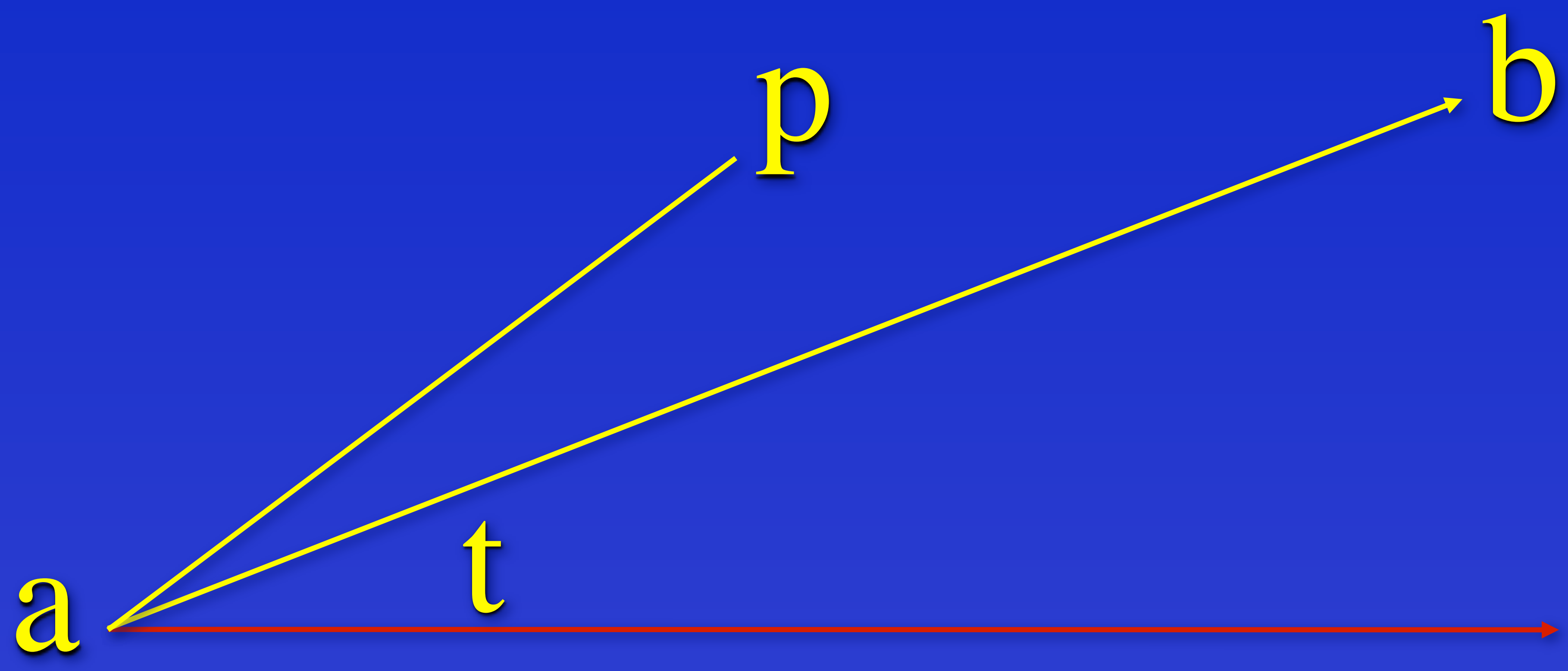
$\sin(t) = (yu - xv) / (u^2 + v^2)$

$\cos(t) = ???$

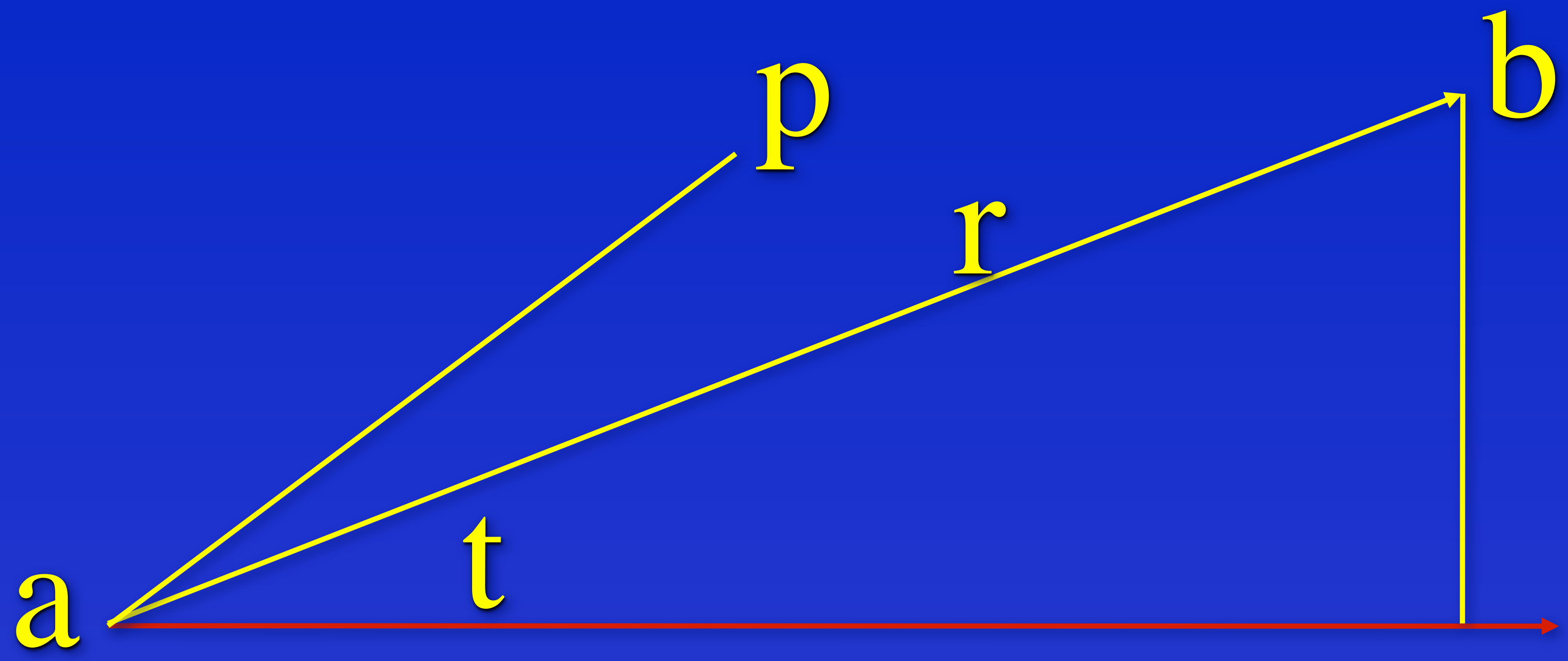
But for a rotation matrix, you don’t need to know t, you just need to know cos(t) and sin(t).

So you don’t need to find t.
You find cos(t) and sin(t) directly.

Is p right or left of a-b?



Rotate p and b to put ab onto the x-axis.

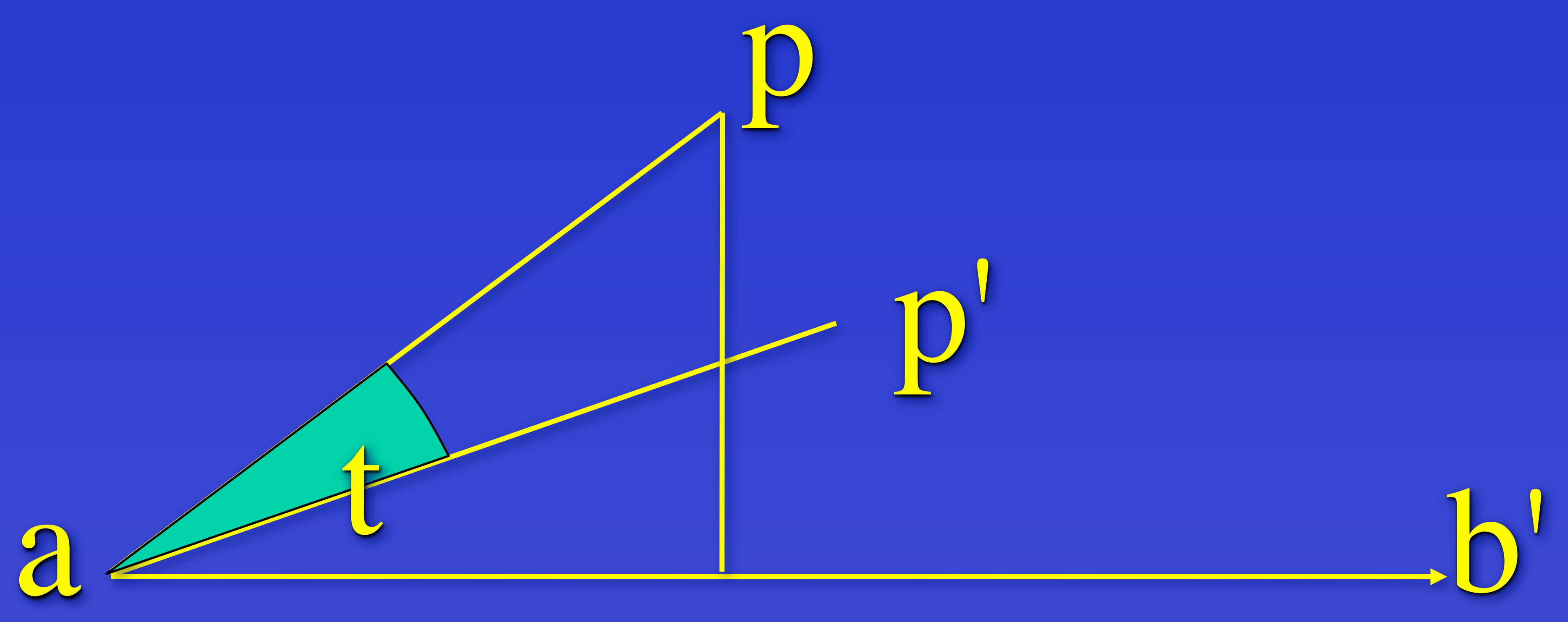


$$\cos(t) = (b_x - a_x)/r$$

$$\sin(t) = (b_y - a_y)/r$$

$$y' = x \sin(-t) + y \cos(-t)$$

$$p'_y - a_y = -(p_x - a_x)(b_y - a_y)/r + (p_y - a_y)(b_x - a_x)/r$$

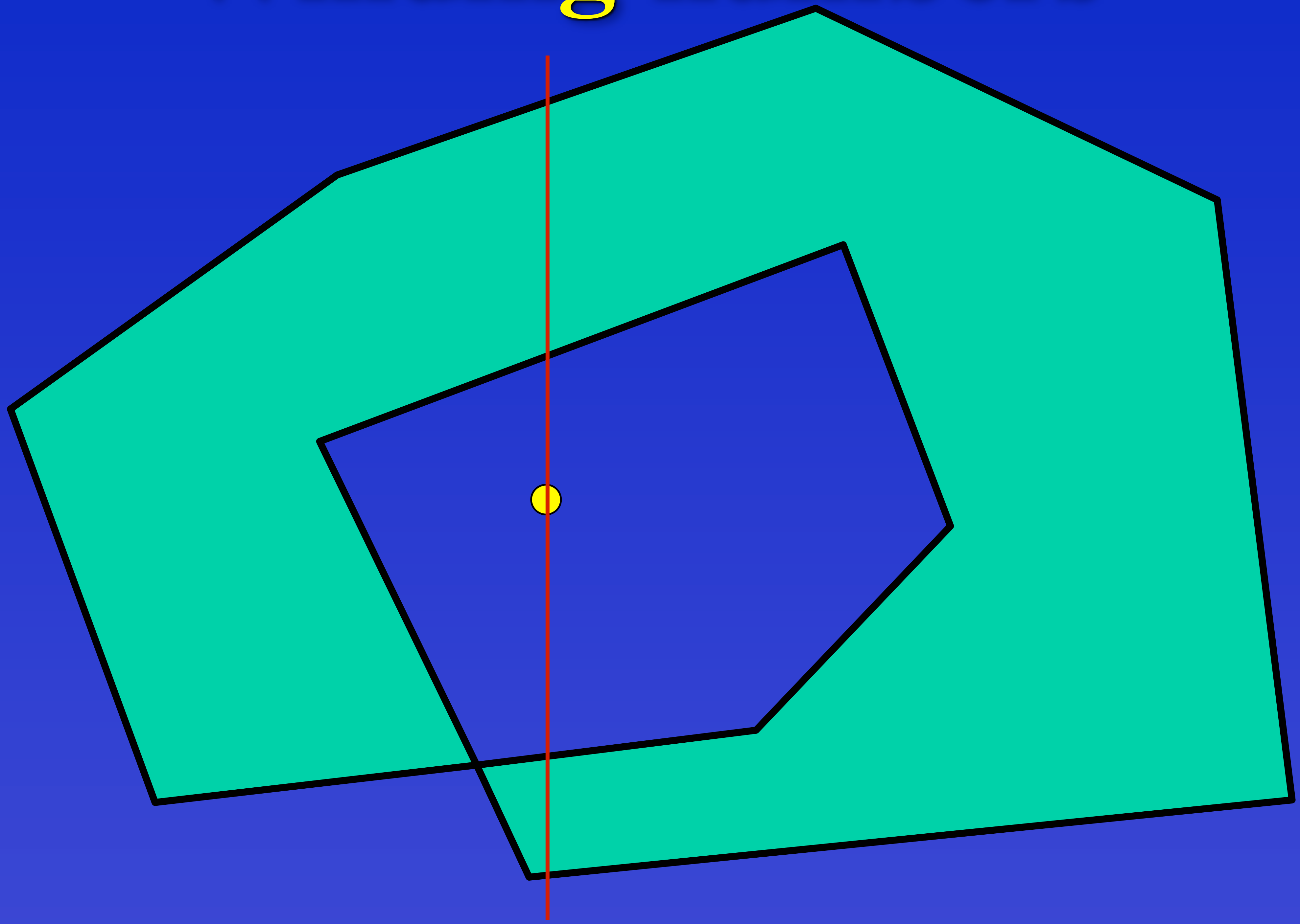


So if $p'_y - a_y > 0$, p is on the left but $r > 0$, so p is on the left iff

$$-(p_x - a_x)(b_y - a_y) + (p_y - a_y)(b_x - a_x) > 0$$

Look: No sin, cos or angles!

Winding numbers



Our winding number w:

- w = left crossings - right crossings
- w = 0 means point is outside
- w = 2 or -2 means point is inside
- w = 4 means point is twice inside (or is that outside?), etc

Alternative definitions

- Winding number is sometimes defined directly (e.g. in textbook):
  - number of times point P is anti-clockwise encircled when tracing around the polygon
- For our fill algorithm, we just need a consistent treatment!