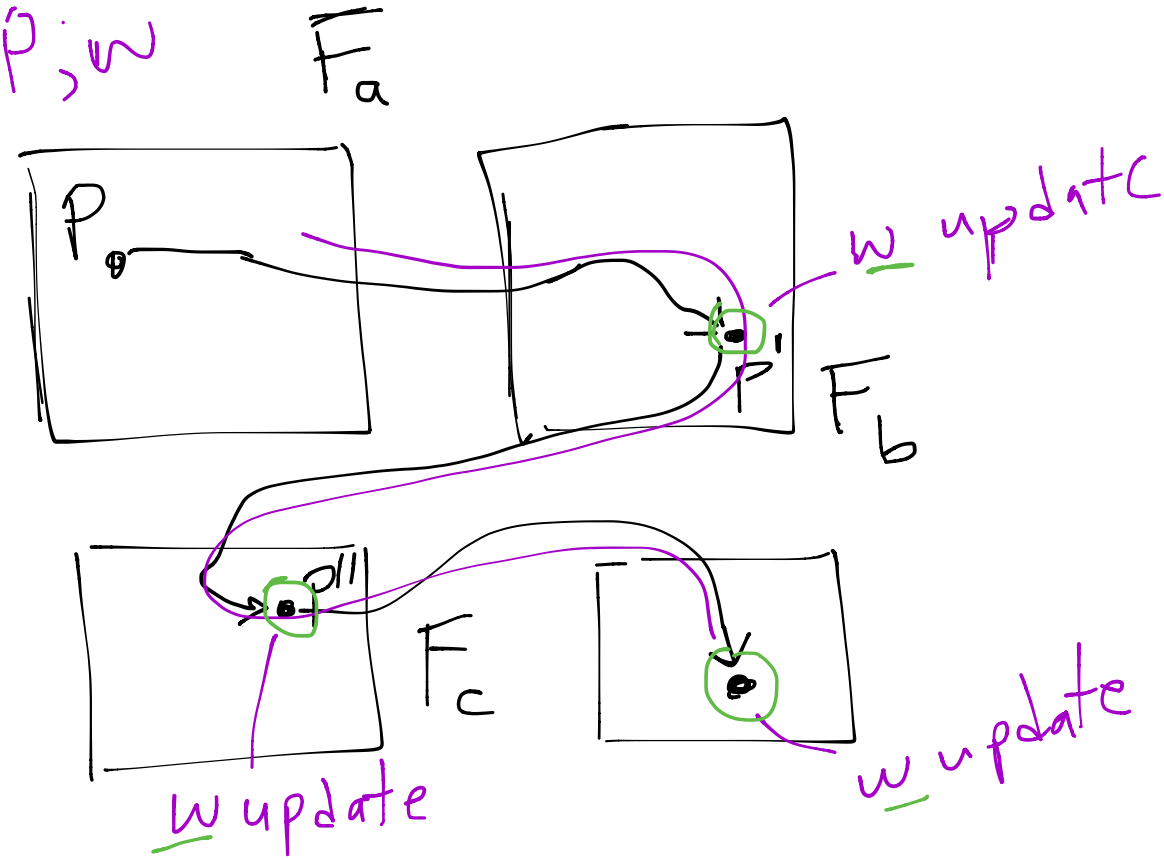
P, w



generate color based on w

update pixel at each P

Initialize

$P = (x, y)$

$-1 \le x \le 1$

$-1 \le y \le 1$

random position

x = random # (-1, 1)

$y = \text{random\#}\ (-1, 1)$

drand48()

$(0, 1)$

$2\,\text{drand48}() - 1$

$(-1, 1)$

random weight

$w = \text{random\#}\ [0, 1]$

Loop until you stop

int i_F = random integer between 0 and N_F

N_F = # of function

$i_F = [\text{drand48}() * N_F]$

$$\rightarrow p = F_{i_F}(p)$$

$$w = \left(w + \text{Weight}(F_i)\right)/2$$

If (loop counter > 20)

Color = LUT(w)

$i = [p.x * N_x]$

$j = [p.y * N_y]$

C_p = pixel color.

image.value(i, j, C_p)

$C_p = ((c_r, c_g, c_b), c_\alpha)$

$c_r * = c_\alpha$

$c_g * = c_\alpha$

$c_p *= c_\alpha$

$c_r += color.r$

$c_g += color.g$

$c_b += color.b$

$c_\alpha += 1$

$c_r /= c_\alpha$

$c_g /= c_\alpha$

$c_b /= c_\alpha$

image.set_value(i, j, c_p)

Back to top of loop