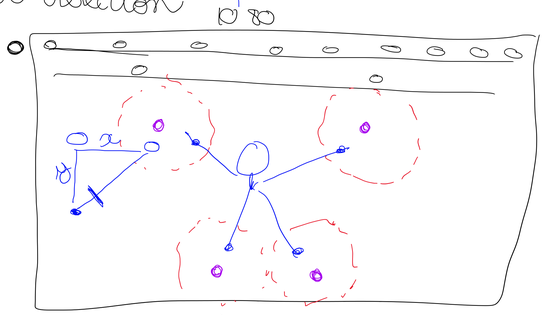
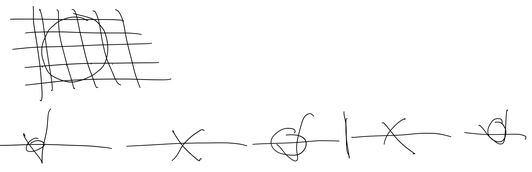
Wednesday, 1 May 2019 15:31

Color detection



rgb(, ,) if (r>x , 0<g<x , 0<b<x



Detect marker ⇒ calculate dist ⇒ map to heat intensity

Color detection. ⇒ pipeline the different stages.

var.: ① RGB 12 bit
② marker position [x, y]

Logic
① Use horizontal edge detection
if current pixel RGB (≠ in_data)

Distance: let x_m, y_m be marker coords
x_0, y_0 be hitzone markers

⇒ $d = \sqrt{(x_m - x_0)^2 + (y_m - y_0)^2}$
⇒ Custom precision, max $D = \sqrt{1920^2 + 1080^2}$

Mapping: $f : d \longmapsto$ heatmap intensity

Roomtemp (Balanced)

Blue
B: 255

Red
R: 255

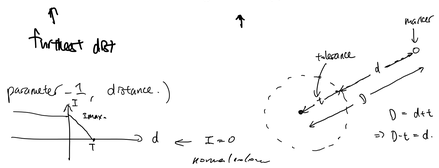
Green = const.

d > T
↑
Furthest dist

d < T
↑

take screen
- map (* out_data, w, h, parameter_1, distance.)
return

Variables
① Distance threshold: 0.1 W.
②



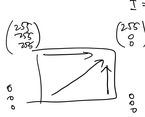
$I = -d + I_{max}$

$I = -\left(\frac{I_{max}}{T}\right) d + I_{max}$.

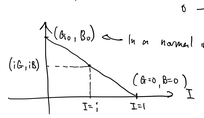
$I = \begin{cases} I_{max}, & d < 0 \\ -\left(\frac{I_{max}}{T}\right) d + I_{max}, & 0 \le d \le T \\ 0, & d > T. \end{cases}$

$p = \begin{pmatrix} R \\ G \\ B \end{pmatrix} + T = Out \begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix} \begin{pmatrix} 255 \\ 0 \\ 0 \end{pmatrix}$

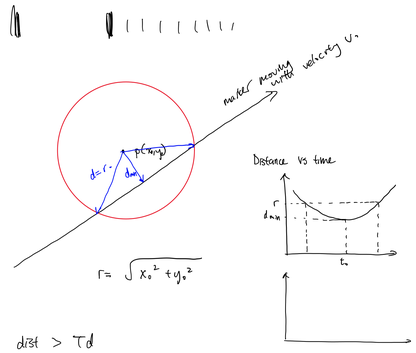
$p = \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \\ -1 \end{pmatrix} = \begin{pmatrix} R \\ G-1 \\ B-1 \end{pmatrix}$



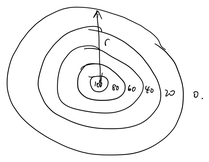
$I = 1 - \frac{d}{threshold}$



I(G, B)

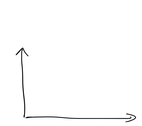


if dist > Td
I 0%
elif d > 0
I variable
else
I 100%

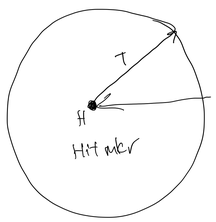


heatmap centered around $p(x_0, y_0)$ with radius r.

for all pixels
if dist from $p(x_0, y_0) > r$
no filter
elif d from p > 0.8 r
0.2 filter
⋮
else 1x filter.



D > T

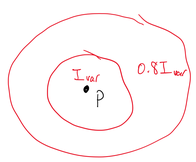


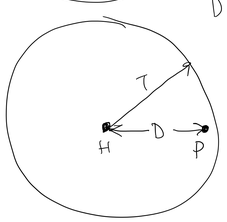
D → p

no filter

player mkr

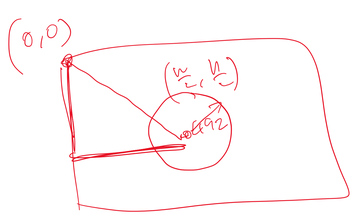
D ≤ T ⇒

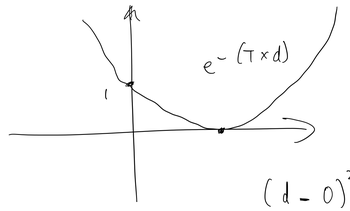




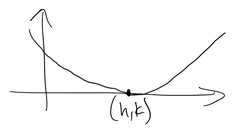
Ivar ⇒ variable I based on

① when D ≤ T, ring filter appears around P with minimum Ivar.
② Calculate Ivar based on D, T
③ As D → 0, I var increases based on RGB transform.



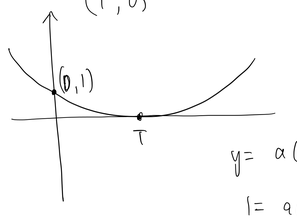


$(d - 0)^2 + k$



$(d-T)^2$

(T, 0)

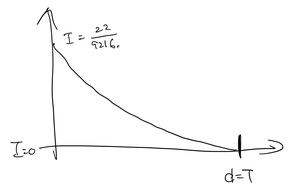


$y = a(x - T)^2$

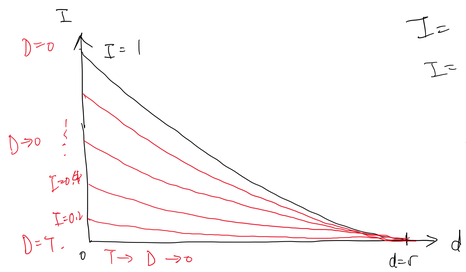
$1 = a(-T)^2$

$\frac{1}{T^2} = a$

$y = \frac{1}{T^2}(x-T)^2$



As $d \to 0$, ring gradient does not change only ring intensity changes.



$I = m d + C.$

$I = \frac{22}{9216} \frac{1}{T} + \frac{22}{9216}$

· As $D \to 0$, $I \to I_{max} = 1$.

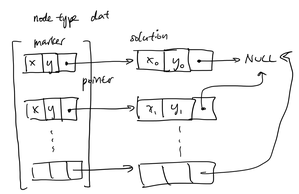
⇒ $I_{max} = \frac{k}{D}$

gradient: $I = \frac{M \cdot I_{max}}{T^2}(d-T)^2$

$I_{max} = 1 - \frac{D}{T}$.

① T → Threshold distance, affects overall intensity of the filter (const.)
② d → independent variable (distance)
③ Imax → Varies proportionally to T.
④ M → External factor, to make filter more observable/responsive.

Multiple inputs:
① How to distinguish between multiple markers?
- Each marker associated to 1 solution.
⇒ 2 arrays
⇒ Map
⇒ Pointers



marker(i) (x_i, y_i) ⇔ solution(i) (P_i, Q_i) array

⇒ returns $dist(i) = \sqrt{(x_i - P_i)^2 + (Y_i - Q_i)^2}$ (R.T)

ring_filter centered around (X_i, Y_i), parameters (... dist(i), const double threshold, const int r, X_i, Y_i)