

So, Question is ... How do I calculate radial Gradient?

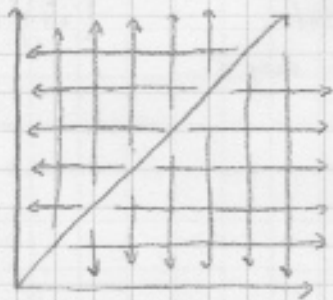
Another idea is to find weights in x , then in y .

Why?

finding
the gradient
will cut it in
by half moving
in direction

If you're trying to find radial gradient of a point, then it'll propagate out in one direction. Making a gradient in one direction. (Change in r)

But in two dimensions, how do I calculate this?



Like in 1 Dimension, we calculate Gradient in y .

But because we have an x component, we calculate it in x too.

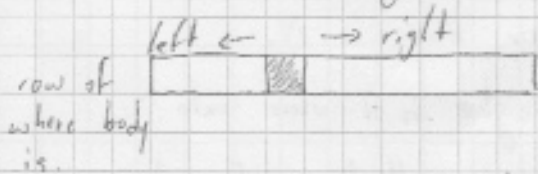
This gives us an \hat{x} & \hat{y} vector, where T believe the resulting magnitude will give us the weight in that direction.

The problem I'm confused about is that the snake is a discontinuous line.

(I think I can arbitrarily
set the adjacent weights
to go down by half.)

Pseudo code:

to calculate x -gradient:



Needs to
happen for every
body tile.

Okay, so I separated
this loop. Not great O time,
but works for now.

- ① we need to get row of where body is.
- ② make a counter that goes down on left & right side of the body file

(while left > 0 and right, lex(body).

if $\text{grid}[\text{left}][\gamma] \cdot \text{weight} \neq 1$ and $\text{left} > 0$

set right of current to previous.
left -- 0

if grid[right][y].weight != 1 and
right < len(body)

set weight of current to previous.
right++