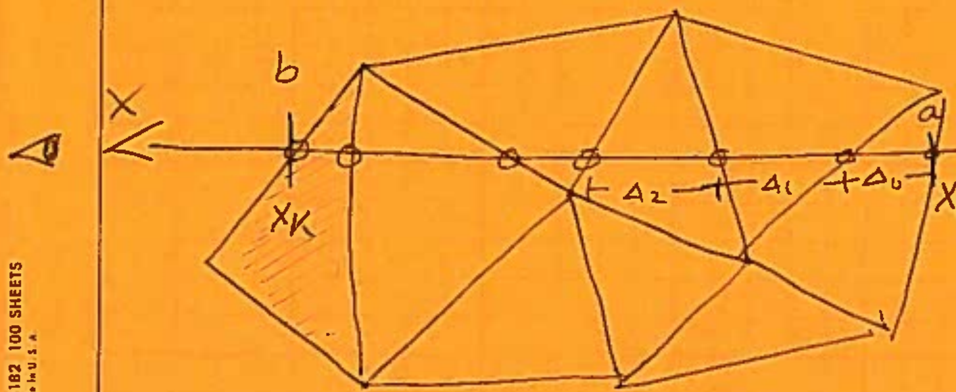
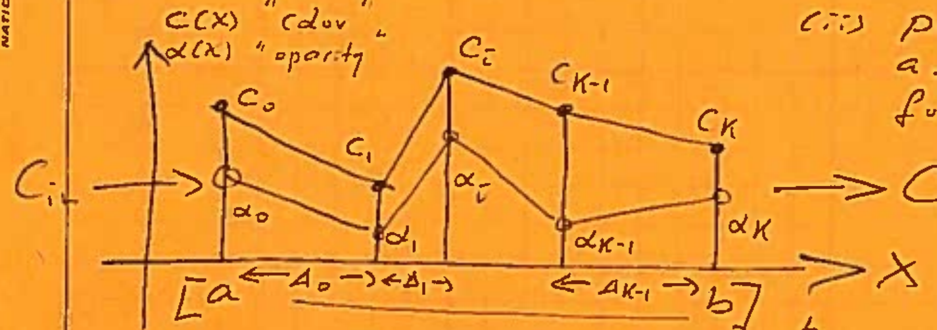
Remark: "Ray casting for unstructured grid / arbitrary sampling"



⇒ need to compute ∩'s between faces & rays...

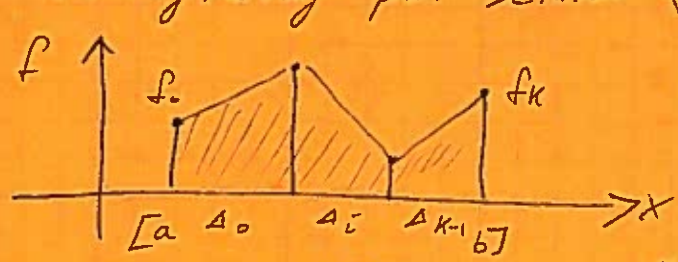
(i) use linear interpolation of fct. values for each Δ
⇒ fct. varies linearly between each consecutive pair of "o"



(ii) perform exact integration along ray using summation for pw. linear functions

Sabella:
$$C_{out} = \int_a^b c(x)\cdot e^{-\int_x^b \alpha(y)\,dy}\,dx + C_{in}\cdot e^{-\int_a^b \alpha(y)\,dy} \qquad (*)$$

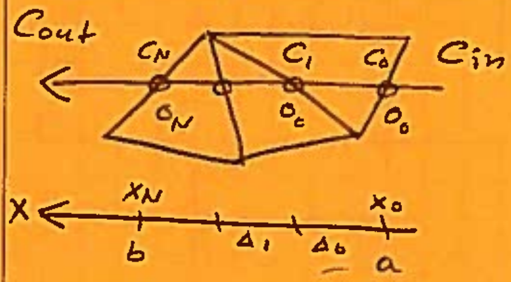
{ integrating pw. linear fct.:



$$\int_a^b f(x)\,dx = \frac{1}{2}\sum_{i=0}^{K-1}\Delta_i\,(f_i+f_{i+1})$$ }

$$\Rightarrow (*) = \frac{1}{2}\sum_{i=0}^{K-1}\Delta_i\left(c_i\cdot e^{-\frac{1}{2}\sum_{j=i}^{K-1}\Delta_j(\alpha_j+\alpha_{j+1})} + c_{i+1}\cdot e^{-\frac{1}{2}\sum_{j=i+1}^{K-1}\Delta_j(\alpha_j+\alpha_{j+1})}\right)$$
$$+ C_{in}\cdot e^{-\frac{1}{2}\sum_{j=0}^{K-1}\Delta_j(\alpha_j+\alpha_{j+1})}$$

More on "Discrete Sabella" -BH- 02-13-1997



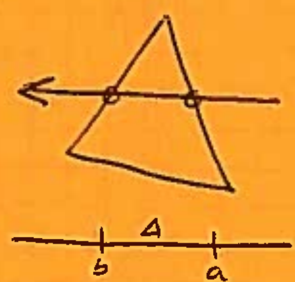
$$C_{out} = \int_a^b col(x) \cdot e^{-\int_{y=x}^{b} opac(y)\,dy}\,dx + C_{in} \cdot e^{-\int_a^b opac(y)\,dy}$$

another $\approx$ approximation

$$\frac{1}{2}\sum_{i=0}^{N-1} \Delta_i \left(c_i + c_{i+1} \right) \cdot e^{-\frac{1}{2}\sum_{j=i}^{N-1} \Delta_j (o_j + o_{j+1})} + C_{in} \cdot e^{-\frac{1}{2}\sum_{i=0}^{N-1} \Delta_i (o_i + o_{i+1})}$$

- **single tetrahedron**

(i) opacity $\equiv 0$
(ii) opacity $\equiv \infty$



- analytical solution

(i) $C_{out} = \int_a^b col(x) \cdot e^0 + C_{in} \cdot e^0$

$= \int_a^b col(x) + C_{in}$ ✓

(ii) $C_{out} = \int_a^b col(x) \cdot e^{-\infty} + C_{in} \cdot e^{-\infty}$

$= 0 + 0 = 0$ ✓

- discrete approximation

(i) $C_{out} = \frac{1}{2} \Delta (c_0 + c_1) \cdot e^0 + C_{in} \cdot e^0$

$= \frac{\Delta}{2}(c_0 + c_1) + C_{in}$ ✓

(ii) $C_{out} = \frac{1}{2} \Delta (c_0 + c_1) \cdot e^{-\infty} + C_{in} \cdot e^{-\infty}$

$= 0 + 0 = 0$ ✓

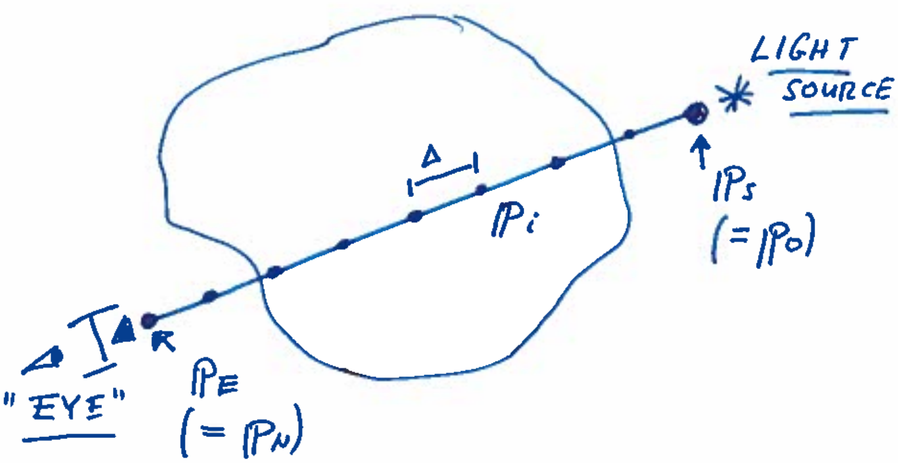
AGREEMENT!

Bernd Hamann
Stratovan

4/4/2016

■ Thoughts on 3D Reconstruction

→ Assumptions / Simplifications:



- Line segment from $\mathbb{P}_S = \begin{pmatrix} x_S \\ y_S \end{pmatrix}$ to $\mathbb{P}_E = \begin{pmatrix} x_E \\ y_E \end{pmatrix}$:

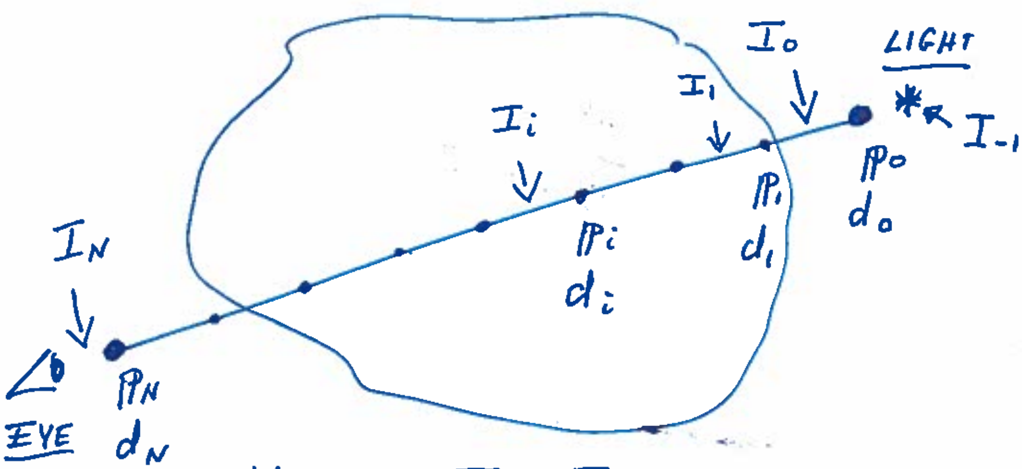
$$\mathbb{P}(t) = (1-t)\mathbb{P}_S + t\,\mathbb{P}_E, \quad t \in [0,1]$$

- Equidistantly placed sample points:

$$\mathbb{P}_i = \mathbb{P}(t_i) = (1-t_i)\mathbb{P}_S + t_i \mathbb{P}_E, \quad t_i = \frac{i}{N}, \quad i = 0 \ldots N$$

- Uniform spacing: $\Delta = \frac{1}{N}$.

→ Assume that a DENSITY Field Must Be Reconstructed:



KNOWN: $I_{-1}, I_N, \mathbb{P}_0, \ldots, \mathbb{P}_N$

UNKNOWN: $d_0, \ldots, d_N$; $d_i \in [0,1]$ "densities"

- Original light intensity: I_{-1}
- Simplified model of absorption by density field $d(x,y)$:

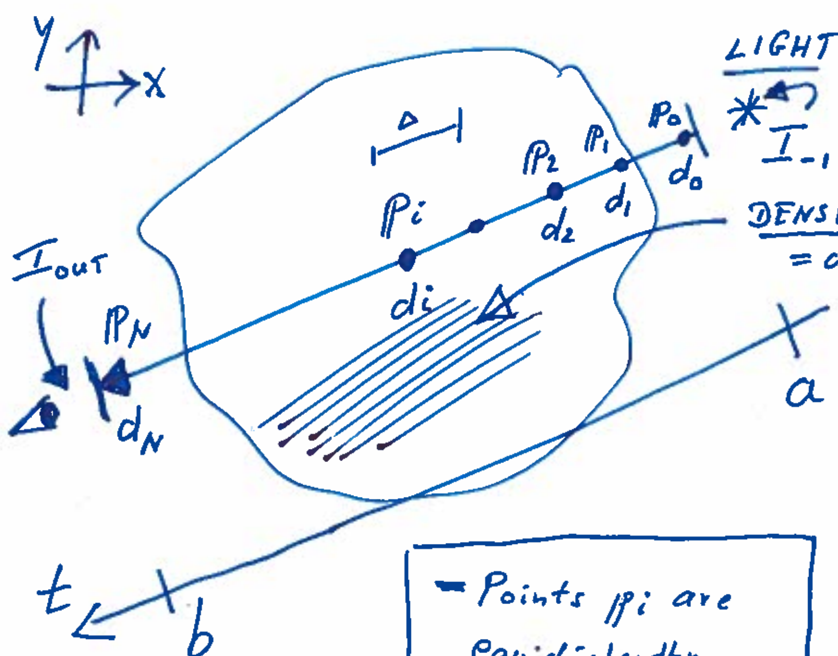
$$I_0 = (1-d_0) I_{-1}$$

$$I_1 = (1-d_1) I_0 = (1-d_1)(1-d_0) I_{-1}$$

$$\vdots$$

$$I_N = \prod_{j=0}^{N} (1-d_j) \cdot I_{-1}$$

→ Simple Absorption Model for Light Passing through Medium of Varying/Unknown Density:



$$\boxed{I_{OUT} = I_{-1} \cdot e^{-\int_a^b dens(t)\,dt}}$$

- Points p_i are equidistantly spaced: constant spacing Δ

⇓ Use TRAPEZOID Rule to Estimate Integral Value:

$$I_{OUT} \approx I_{-1} \cdot e^{-\sum_{i=0}^{N-1} \frac{\Delta}{2}(d_i + d_{i+1})}$$

⇓

$$\frac{I_{OUT}}{I_{-1}} = e^{-\sum_{i=0}^{N-1} \frac{\Delta}{2}(d_i + d_{i+1})}$$

⇒ Apply logarithm on both sides:

$$\ln\left(\frac{I_{OUT}}{I_{-1}}\right) = -\frac{\Delta}{2} \sum_{i=0}^{N-1} d_i + d_{i+1}$$

$$-\frac{2}{\Delta} \ln\left(\frac{I_{OUT}}{I_{-1}}\right) = \sum_{i=0}^{N-1} d_i + d_{i+1}$$

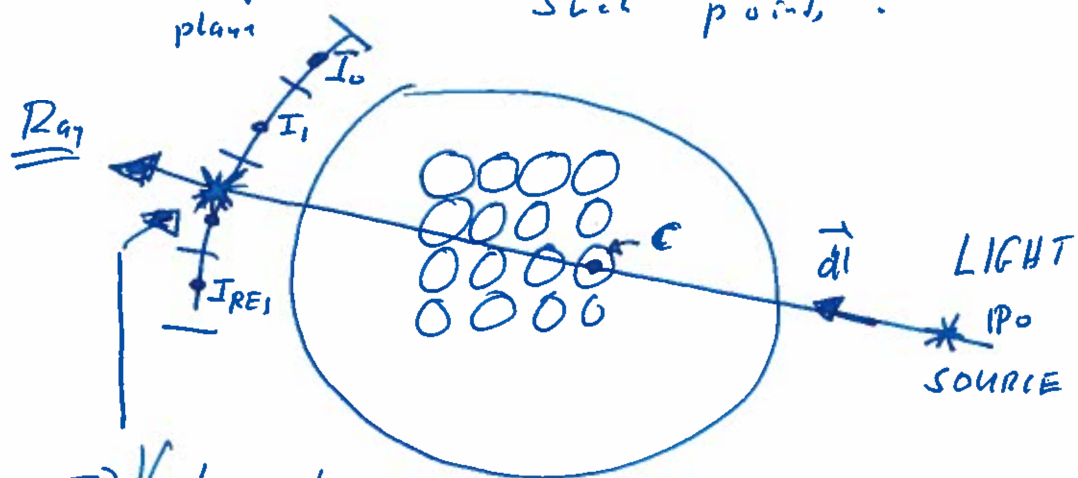
$$= d_0 + 2d_1 + 2d_2 + \ldots + 2d_{N-1} + d_N$$

$$= [1 \;\; 2 \;\; 2 \;\ldots\; 2 \;\; 1] \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_N \end{bmatrix}$$

⇒ ONE linear equation for (N+1) unknown values $d_0, \ldots, d_N$ BH
⇒ Consider additional rays to define an over-determined system. ■

→ Want to "reconstruct" values for specific points / cells / voxels in 3D space

⇒ must force / select rays to go through such points?



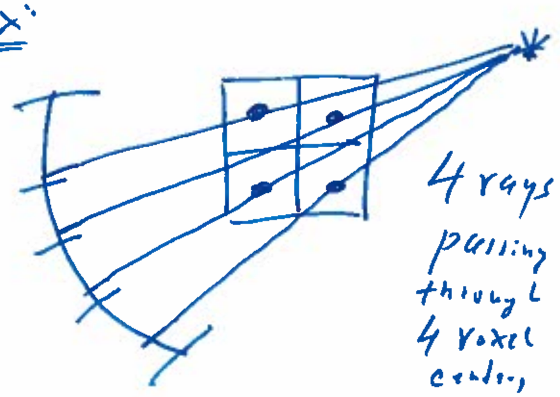
⇒ Value at location "*" not an originally recorded pixel intensity value

⇒ $\boxed{\text{Intensity } (*) = \text{Interpolated Intensity From Pixel Neighborhood}}$

⇒ Ray direction: $\boxed{\vec{dl} = C - P_0}$

⇒ Can force rays to pass through all centers C being voxel / region centers in 3D space:

Ex:

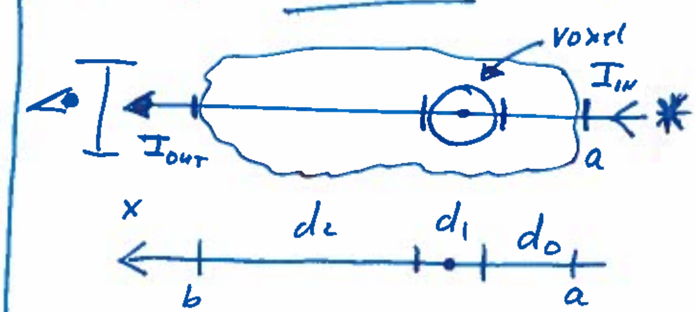


4 rays passing through 4 voxel centers

e.g., 512^3 voxels

⇒ need $\geq 512^3$ lin. independent "conditions" / equations for the unknown voxel densities!

→ Consider SINGLE Voxel situation:



d_i: average densities over the 3 spans

$$I_{OUT} = I_{IN} \cdot e^{-\int_a^b d(x)\,dx}$$

density

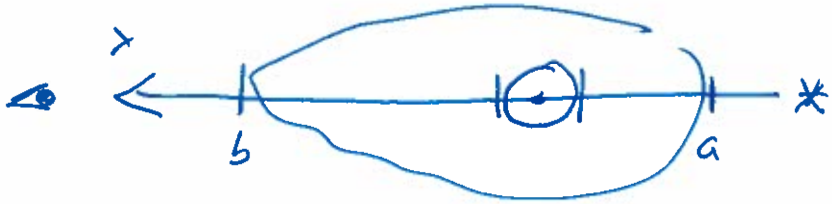
⟹ Cont'd.

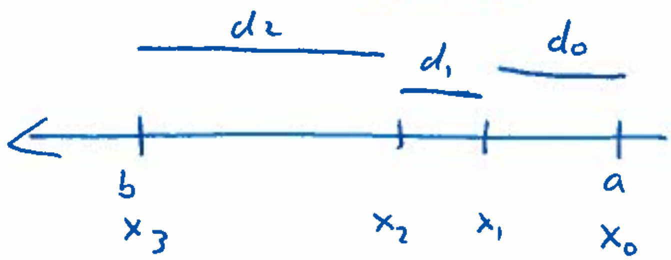
— Bernd Hamann
Stratovan

- BH -

4/11/2016

⇒ LOCAL, SINGLE VOXEL SITUATION:





⇐ pw. constant density fct. over 3 spans

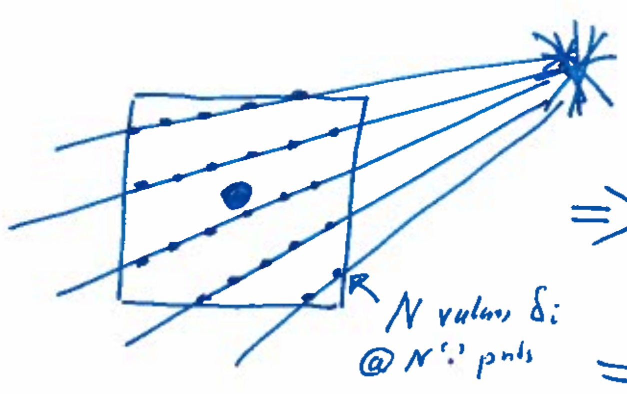
⇐ $\int_a^b dens(x)\, dx$

$$\approx d_0(x_1-x_0) + d_1(x_2-x_1) + d_2(x_3-x_2) = \sum_0^2 d_i \cdot (x_{i+1}-x_i)$$

- **BUT:** NOT ONLY IS d_1 UNKNOWN BUT d_0, d_2 are unknown as well...

■ IDEA 2

"MONTE CARLO" method: CONSIDER MANY ray samples inside one voxel – then combine the densities at these samples "properly" to define the unknown average density of the voxel:



N values δ_i @ N pnts

$\Rightarrow dens(\bullet)$ = average Of Dens At All (·)

$$\Rightarrow dens(\bullet) = \frac{1}{N}\sum_1^N \delta_i \;???$$

■ IDEA 3

→ Meshless Method: Reconstruct density values at scattered points in a pnt. cloud

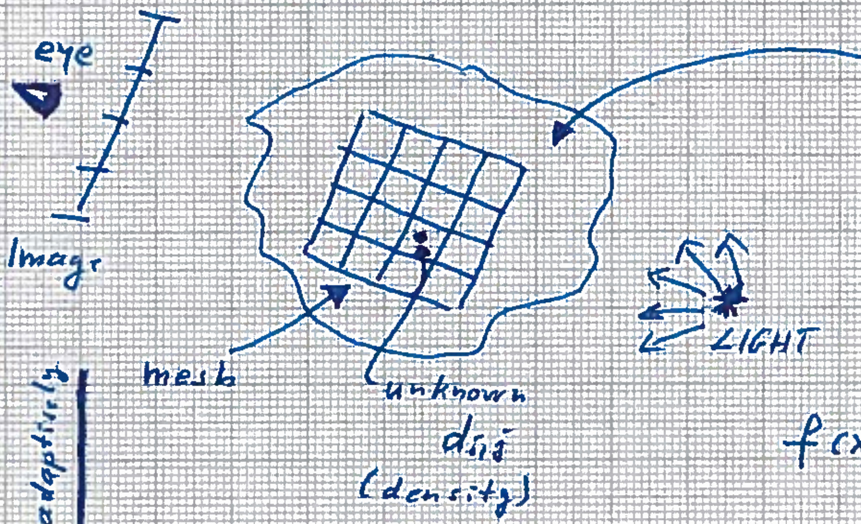
~BH

Bernd Hamann
STRATOVAN

4/18/2016

3D Reconstruction: More Issues & Approaches

(i) Differential Equation and Finite Element Approach



Field ("density") function is a continuous function approximated via a linear combination for a finite element method:

$$f(x,y) = \sum_{i,j} c_{i,j} \cdot H_{i,j}(x,y) \;!$$

e.g., $H_{i,j}(x,y)$ are BOX functions over each voxel...

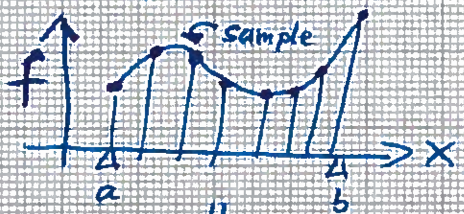
⇒ Set up linear system for unknown coefficients $c_{i,j}$ (ideally over-determined)

⇒ Can set up problem to compute piecewise CONSTANT, LINEAR, QUADRATIC, CUBIC, ... approximation of f.

OR: USE RADIAL BASIS FUNCTION METHOD WITHOUT A MESH...
$f = \sum_i c_i R_i(x,y)$
⇒ choose R_i adaptively

(ii) On Approximating LINE INTEGRALS

• General context:

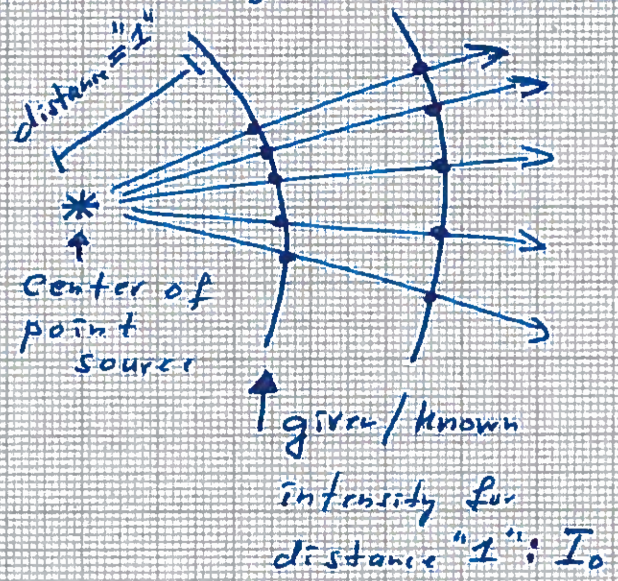


⇓ CONTEXT

Can use different-degree polynomial approximations of f (locally) to estimate

$\int_a^b f(x)\,dx$:
a) piecewise (pw) CONSTANT
b) pw. LINEAR (trapezoid rule)
c) pw. QUADRATIC (Simpson rule)

(iii) Point Light Source and ATTENUATION



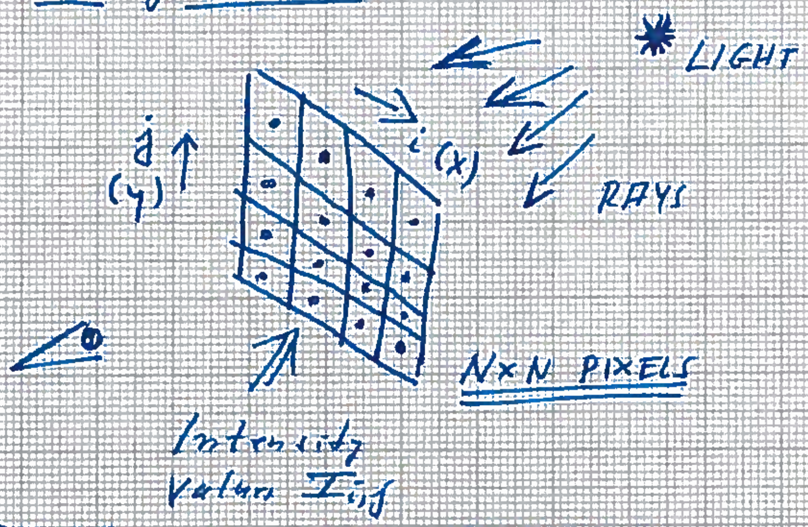
⇒ Photon/ray density decreases QUADRATICALLY with increasing distance:

$$I(dist) \sim \frac{1}{dist^2} \quad !$$

$$I = I_0 / dist^2$$

$$\left(\Rightarrow \text{more general: } I = \frac{I_0}{A + B \cdot dist + C \cdot dist^2} \right)$$

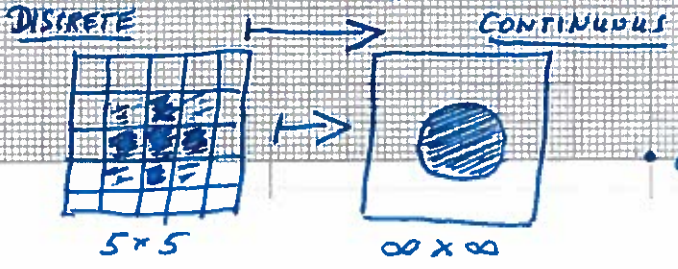
(iv) Possibility of Computing and Using a CONTINUOUS/SMOOTH Function $F(x,y)$ in Image Plane.



⇒ IDEA:

- Interpolate the N×N pixel intensities with a smooth bi-cubic B-spline function $F(x,y) =$

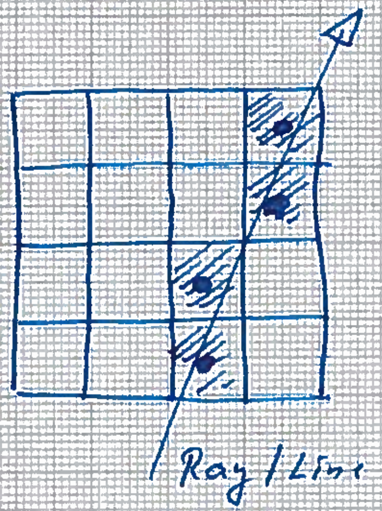
$$= \sum_i \sum_j d_{i,j} N_i^3(x) N_j^3(y)$$



- CAN EVALUATE $F(x,y)$ USING ANY HIGH RESOLUTION!

(v) BRESENHAM'S Algorithm for Voxel Grid

2D Bresenham:



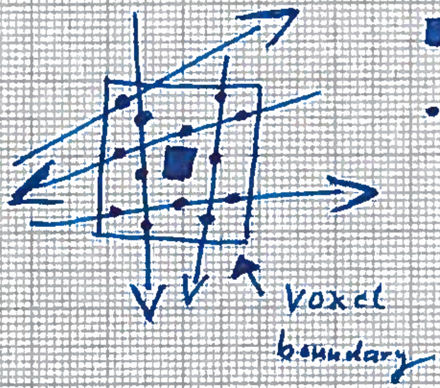
⇒ "Turn on those pixels (→ voxels) whose centres are closest to the line ..."

⇒ Can generalize 2D Bresenham to 3D case !

⇒ Can generate a 'sequence' of voxels 'connected to each other' defining best voxel set approximation of line/ray...

(vi) 'Simplification' – Allowable/Good ??? :

Identify ray samples inside a voxel with voxel centre



■ voxel centre

• sample locations on rays passing through voxel

⇒ Assuming that the • locations are nearly uniformly distributed:

$$\text{density}(\blacksquare) = \frac{1}{k}\sum_i \text{dens}_i \quad , \quad !$$

where $\text{dens}_i = \text{density}(\bullet_i)$

~BH