

- initializeGL()
- resizeGL()
- paintGL()
- mousePressEvent()
- mouseReleaseEvent()

all in gltexobj.cpp

(alg 05) + (lab 11)
Qt vis. Kd-Tree

glwinobj.cpp

- for the GL window
set :

texwin → set Mouse Tracking
(true);

- Create file, edit
menus ✓ ↓
read in kdtree
prints. in

gl_texture.cpp :

(GL window, contains

double x, y : input point
(more)

double x1, x2,
y1, y2 : var box

vector<point_t> pt1;

KdTree_t<point_t, point_t*,
point_t> kdtr

point_t *
point_t *
double

query
knearest
radius

int

k

vector<point_t * > knearest

vector<point_t * > range

Initialize GLC):

glClearColor(0, 0, 0, 0)

Resize GLC):

glViewport(0, 0, w, h)

glMatrixMode(GL_PROJECTION)

glLoadIdentity()

gluOrtho2D(-w, w, -h, h)

glMatrixMode (GL_MODELVIEW)
glLoadIdentity ()

paintGL ()

// draw main point

// draw input points
(if any)

// draw kd-tree
(if not empty)

// draw query point

// draw values

// nearest point(s)

// bounding box.

// & nearest points

glDrawBuffer (GL_BACK)

glClear (GL_COLOR_BUFFER_BIT)

GL_DEPTH_BUFFER_BIT);

I draw with box

glColor4f(1,1,1,1)

glBegin(GL_QUADS)

glVertex(x-5, y-5);

|| (x+5, y-5)

|| (x+5, y+5)

|| (x-5, y+5)

glEnd();

- drawing circle:

glPushMatrix()

glTranslatef(x, y, z)

gluDisk(0.0, 0.0, *

of grey pt

(*grey)[0],

(*grey)[1],

gluDisk(0.0, 0.0, radius=2.0,
radius=360, 1)

glPopMatrx()

⋮

range of y : a

glRectd(x_1, y_1, x_2, y_2)

glPolygonMode(GL_FRONT_

AND_BACK, GL_LINE)

see this (was to see pt 111)

- MouseEvent (MouseEvent & e)

MapCoords(e → x(), e → y())
ex ey

let xobj :: x = ex;

let yobj :: y = height() - ey;

double ex = (double)x / width;

$$\text{double } dy = (\text{double}) y / \text{height}()$$

$$dx = \text{width}() * (2.0 * dx - 1.0)$$

$$dy = \text{height}() * (2.0 * dy - 1.0)$$

$$x = dx;$$

$$y = dy;$$

}

if (e->button() ==
Qt::LeftButton)

if (e->modifiers() ==
Qt::NoModifier)

pts.push_back(
new point_t(x, y))

else

i) $(e \rightarrow \text{modifiers}) = =$
 $Q + :: \text{Alt Modifier}$
// never hypo.

given
 \rightarrow do nothing.

the code in Release
Event is kept

i) $\text{Ext} :: \text{Shift} / \text{A Mod} / \text{e}$

$$x_1 = x$$

$$y_1 = y$$

$$x_2 = x$$

$$y_2 = y$$

make the hand event (...)

— cells

ktmo.un(gvey,

heart, radius)

(if Alt Mode/e)

— if Shift Modifier :

$$x2 = x$$

$$y2 = y$$

Movie Mon Event :

map Goods ($e \rightarrow x()$
 $e \rightarrow y()$)

$x2 = x$

$y2 = y$

$e \rightarrow accept()$

update $64()$ // add/w

- kd tree. render() :

- render in in order

render($t \rightarrow \text{left}$)

// draw self

render($t \rightarrow \text{right}$)

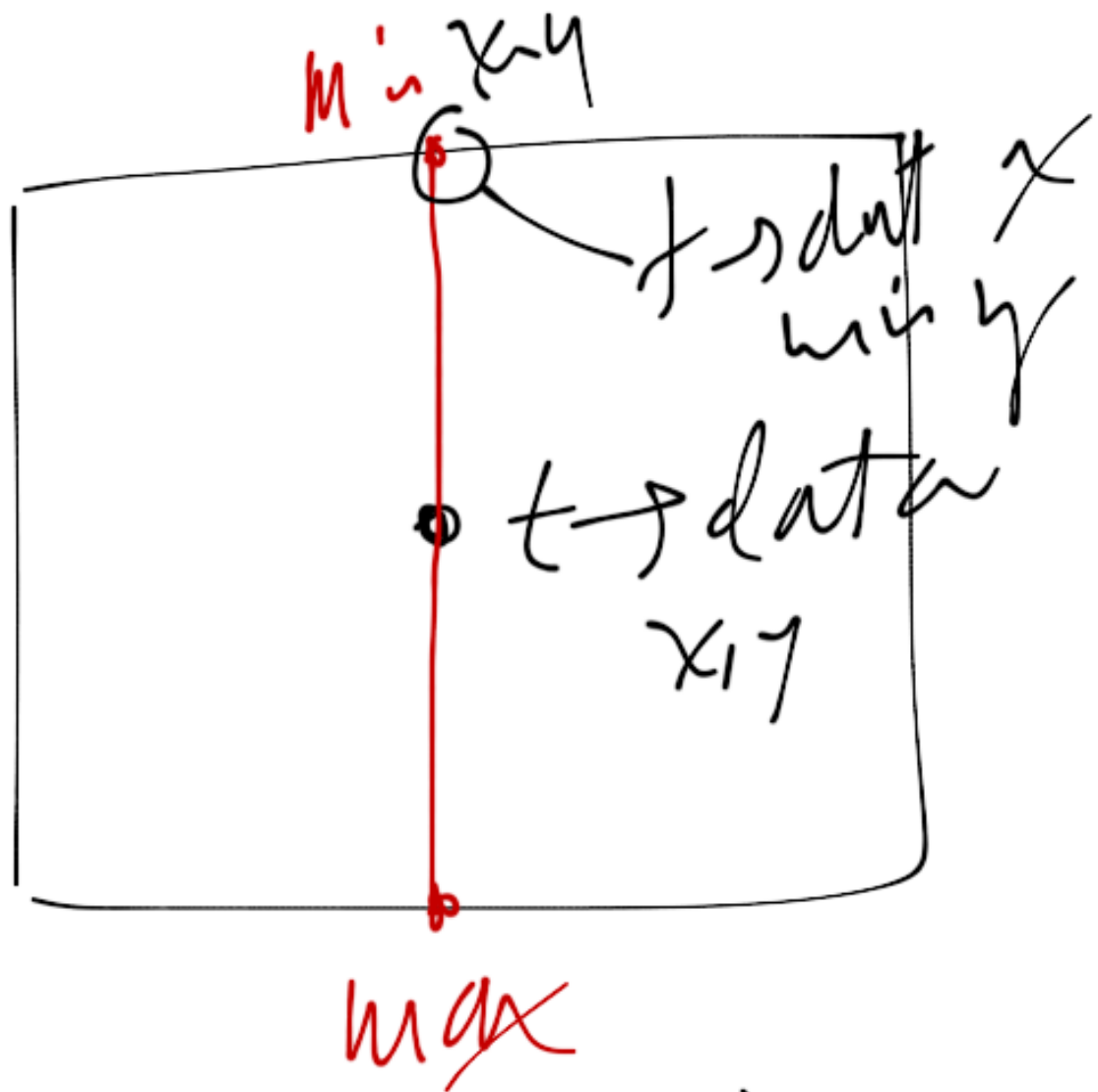
- drawing Kd tree node:

- figure out which axis (horiz or vert. line)

- draw line (min, max, P_{val})
 x, y

∴ draw lines

line (P_{val} , max)
 x, y



Basic (GL-LINES)

glVertex2f (t -> data)[0],

(t -> data)[1])

glVertex2f (t -> data)[0], t -> min[1])