登录/注册 会员中心 🎁 收藏 动态



linux图像显示

linux图像显示(一)framebuffer操作

linux图像显示 (二) bmp图片

linux图像显示 (三)使用libjpg处理jpg图片

linux图像显示 (四)使用libpng处理png图片

linux图像显示 (五)使用freetype处理矢量字体

FreeType库是一个完全免费(开源)的、高质量的且可移植的字体引擎,它提供统一的接口来访问多种字体格式文件,包括 TrueType, OpenType, Type1, CID, CFF, Windows FON/FNT, X11 PCF等。

下面讲解怎么使用freetype,包括freetype的移植,freetype的使用实例

库和示例代码: https://download.csdn.net/download/weixin 42462202/10855217

1、移植库

(1)在ubuntu下解压,进入源码目录

(2)配置

①先指定编译工具链

env CC=arm-linux-gcc

②指定运行平台和库安装目录,这里我的安装目录是/opt/mlib

./configure --host=arm-linux --prefix=/opt/mlib

③编译安装

1 make

2 make install

现在在/opt/mlib目录下就生成了头文件和库

2、freetype的使用

(1)在项目中使用库

库现在已经在/opt/lib目录下,怎么使用呢?

答:通过-I-L-I给编译器指定头文件路径和库

-1:指定头文件路径

目录	\nearrow
linux图像显示 1、移植库	
2、freetype的使用	
分类专栏	\nearrow
C++ C/C++	1篇
STL源码剖析	18篇
MySQL	4篇
】 Linux内核	14篇
分布式	
nginx	
从零开始写一个RTSP	10篇
live555源码分析与应用	9篇
人 Linux驱动	19篇

-1 :指定库路径

-1:指定库

为了方便,写一个简单的Makefile

```
CFLAGS += -I/opt/mlib/include/ -I/opt/mlib/include/freetype2/ -I/opt/mlib/include/freetype2/freetype/ -I/opt/mlib.
LDFLAGS += -L/opt/mlib/lib/
LDFLAGS += -lfreetype -lm

all:
    arm-linux-gcc examplet.c -o example $(LDFLAGS) $(CFLAGS)
```

(2)freetype使用步骤说明

①初始化库

```
1 | FT_EXPORT( FT_Error )
2 | FT_Init_FreeType( FT_Library *alibrary );
```

②创建face 对象来打开字体文件

③选择字符的大小

```
pixel_width:字体的像素宽度
pixel_height:字体的像素高度

FT_EXPORT(FT_Error)
FT_Set_Pixel_Sizes(FT_Face face,
FT_UInt pixel_width,
FT_UInt pixel_height);
```

④添加渲染(可选,斜体、加粗、下划线等)

```
1  FT_EXPORT( void )
2  FT_Set_Transform( FT_Face face,
3  FT_Matrix* matrix,
4  FT_Vector* delta );
```

⑤转化bitmap位图

⑥开始绘制

```
1  void
2  draw_bitmap( FT_Bitmap* bitmap,
3  FT_Int  x,
4  FT_Int  y)
```

(3)freetype使用示例

①在控制台显示

```
1 | #define WIDTH 640
 2 #define HEIGHT 480
 4 unsigned char image[HEIGHT][WIDTH];
 6 int main(int argc, char **argv)
7 {
 8
       FT_Library library;
9
       FT_Face
                    face;
10
11
       FT_GlyphSlot slot;
12
       FT_Matrix
                    matrix;
                                         /* transformation matrix */
13
       FT_Vector
                                         /* untransformed origin */
                    pen;
14
       FT_Error
                    error;
15
16
       char*
                    filename;
17
       char*
                    text;
18
19
       double
                    angle;
20
                    target_height;
21
       int
                    n, num_chars;
22
23
       if ( argc != 3 )
24
25
           fprintf ( stderr, "usage: %s font sample-text\n", argv[0] );
26
           exit( 1 );
27
28
29
       filename
                   = argv[1]; /* 获取字体格式文件 */
30
       text
                   = argv[2]; /* 获取显示内容 */
31
                   = ( 25.0 / 360 ) * 3.14159 * 2; /* 字体旋转角度 */
       angle
32
33
       target_height = HEIGHT; /* 得到屏幕的高度 */
34
35
       /* 1.初始化库 */
36
       error = FT_Init_FreeType( &library );
37
38
       /* 2. 创建一个新的 face 对象来打开一个字体文件 */
39
       error = FT_New_Face( library, argv[1], 0, &face );
40
41
       /* 3.选择一个字符大小 */
42
       error = FT_Set_Pixel_Sizes(face, 24, 0);
43
44
       /* 字体旋转 */
45
       angle = (25 / 360 ) * 3.14159 * 2; /* 旋转25度 */
46
       matrix.xx = (FT_Fixed)( cos( angle ) * 0x10000L );
       matrix.xy = (FT_Fixed)(-sin( angle ) * 0x10000L );
```

```
48
        matrix.yx = (FT_Fixed)(sin(angle)*0x10000L);
49
        matrix.yy = (FT_Fixed)( cos( angle ) * 0x10000L );
50
51
        /* 图像绘制起点 */
52
        pen.x = (g_var.xres / 2 + 8 + 16) * 64;
53
        pen.y = (g_var.yres / 2 + 8 + 16 - 35) * 64;
54
55
        for (n = 0; n < num chars; n++)
56
57
            /* 4. 渲染字体 */
58
            FT_Set_Transform( face, &matrix, &pen );
59
60
            /* 5.转化bitmap位图 */
61
            error = FT_Load_Char(face, text[n], FT_LOAD_RENDER);
62
63
            /* 6.开始绘制 */
64
            draw_bitmap( &slot->bitmap,
65
                        slot->bitmap_left,
66
                        target_height - slot->bitmap_top );
67
68
            pen.x += slot->advance.x;
69
            pen.y += slot->advance.y;
70
71
            show_image();
72
73
74
        FT_Done_Face ( face );
75
        FT_Done_FreeType( library );
76 }
77
78
79
    draw_bitmap( FT_Bitmap* bitmap,
80
                   FT_Int
81
                   FT_Int
                              y)
82
83
        FT_Int i, j, p, q;
84
        FT_Int x_max = x + bitmap->width;
85
        FT_Int y_max = y + bitmap->rows;
86
87
88
        for ( i = x, p = 0; i < x_max; i++, p++ )
89
90
            for ( j = y, q = 0; j < y_max; j++, q++ )
91
92
               if ( i < 0 || j < 0
93
                      i >= WIDTH || j >= HEIGHT )
94
                   continue;
95
96
                image[j][i] |= bitmap->buffer[q * bitmap->width + p];
97
98
99
100
101 void
102 | show_image( void )
103
104
        int i, j;
105
106
        for ( i = 0; i < HEIGHT; i++ )</pre>
107
108
        for (j = 0; j < WIDTH; j++)
```

测试方法:

②在Icd上显示

在lcd上显示只需要修改这个函数

```
1 void draw_bitmap( FT_Bitmap* bitmap, FT_Int x, FT_Int y)
2 {
3
      FT_Int i, j, p, q;
       FT_Int x_max = x + bitmap->width;
       FT_Int y_max = y + bitmap->rows;
8
       for ( i = x, p = 0; i < x_max; i++, p++ )
9
10
          for ( j = y, q = 0; j < y_max; j++, q++ )
11
              if ( i < 0 || j < 0 ||
12
13
              i >= g_var.xres || j >= g_var.xres )
14
              continue;
15
16
              /* 将这句话改为在Lcd的指定位置打点 */
17
              //image[j][i] |= bitmap->buffer[q * bitmap->width + p];
18
              lcd_put_pixel(i, j, bitmap->buffer[q * bitmap->width + p]);
19
20
21 }
```

③显示中文

可以通过改变FT_Load_Char(face, text[n], FT_LOAD_RENDER);的传参来显示中文

```
1 | wchar_t *chinese_str = L陪陪啥";
2 |
3 | error = FT_Load_Char(face, chinese_str[n], FT_LOAD_RENDER);
```

在 linux C 环境下wchar_t表示一个字符4个字节



从零开始写一个RTSP服务器 (—) RTSP 协议讲解 ◎ 47079

C++ deque的用法与示例 ① 27119

Linux I2C驱动框架(超详细) @ 26351

我的开源项目-RtspServer © 20913

从零开始写一个RTSP服务器 (二) RTSP 协议的实现 ⊙ 17619

最新评论

A

live555源码分析(二)基本组件上 abcd552191868: 你这代码是不全呢?还 是版本很老了?

我的开源项目-RtspServer

李长歌、: 生成下core_dump文件看下堆栈信息, 排查下

我的开源项目-RtspServer

李长歌、: 生成下core_dump文件看下堆栈信息, 排查下

160,4900

Linux I2C驱动框架(超详细)

linux下用freetype2显示汉字

18-23

本人在工作实践中写的一个汉字显示程序,在linux平台下,利用frambuffer显示设备来显示。用freetype2库来显示矢量汉字。程序带有汉字对齐、加粗、...

00

请发表有价值的评论,博客评论不欢迎灌水,良好的社区氛围需大家一起维护。



评论

🥙 幽灵之箭: 兄弟,变量g_var在哪定义的啊 9月前 回复 •••

d

数码相框(四、使用freetype库实现矢量字体显示)_Louis@...

11-5

注:本人已购买韦东山第三期项目视频,内容来源《数码相框项目视频》、数码相框-通过freetype库实现矢量显示,只用于学习记录,如有侵权,请联系删除。

Linux的freetype矢量字体库安装_u010832976的博客

10-25

freetype是矢量字体的库文件,我们要先得到freetype的压缩包,解压缩后进入目录,输入./configure --host=arm-linux --prefix=/work/tools/gcc-3.4.5-glibc-2.3....

Linux电子书项目之freetype实现矢量字体的显示(2)

DaLaoZi的专栏 @ 360

上一篇博文我们通过点阵数组实现了字母的显示: http://blog.csdn.net/shenhuan1104/article/details/79476053现在我们通过freetype这个框架在PC和单...

linux上freetype2安装编译和使用 热门推荐

dongtinghong的专栏 ② 2万+

目前正在做linux上的opengl开发,是做视频<mark>图像处理</mark>的,需要叠加一下文字动画和特效。用opengl的做法是先生成文字贴图然后用opengl渲染在视频帧上...

linux应用项目(一)数码相框(2)数码相框之freetype实现矢量字体的...

7-31

现在我们通过freetype这个框架在PC和单板LCD上分别实现**矢量字体**的显示。一、PC上**矢量字体**的显示(1) 矢量字体:存储一些关键点,然后通过贝塞尔曲...

LINUX学习-在LCD上显示一个矢量字体 vogach的博客

11-3

本章需要做的事情就是在LCD上显示一个矢量字体第一步安装freetype库 1、freetype文件库的使用将freetype-2.4.10.tar.bz2拷贝到linux虚拟机后,使用tar...

基于嵌入式linux的freetype矢量字体简单显示的实现

TECH PRO的博客 ① 2412

一、freetype简介 FreeType库是一个完全免费(开源)的、高质量的且可移植的字体引擎,它提供统一的接口来访问多种字体格式文件,可以非常方便我...

freetype在Linux平台编译小记

SP的去栏 **△** 3367

一、准备工作: 下载freetype , 地址:http://www.freetype.org/download.html , 本文所用版本:2.4.10 下载zlib , 地址:http://www.zlib.net/ , 本文所用版本...

Linux下,利用FreeType2的API实现字符的显示

你访问的页面已存在 ⊙ 8173

网上的FreeType2例子太少,能显示汉字的比较难找,C语言代码写的更难找,能找到的,基本上是被转载了N遍的同一个示例代码,基本上解决不了我的...

Linux学习笔记(二十) -- 在Ubuntu上进行FreeType库的编译

以文会友 ① 3335

1.FreeType库 FreeType库是一个完全免费(开源)的、高质量的且可移植的字体引擎,它提供统一的接口来访问多种字体格式文件,包括TrueType,Ope...

linux C读取字库文件

₩₩水的捕突 ○ 2406

当叫众人知道你们谦让的心。——腓立比书【4:4】前段时间在Linux下用C写了一个读取字库的代码,我感觉有必要记录一下。 #include <stdio.h> #inclu...

linux图像显示 (三)使用libjpg处理ipg图片

JT同学的博客 **◎** 2439

linux图像显示 linux图像显示 (一) framebuffer操作 linux图像显示 (二) bmp图片 linux图像显示 (三) 使用libjpg处理pg图片 linux图像显示 (四) 使用libp...

linux图像显示(四)使用libpng处理png图片

JT同学的博客 **⊙** 2370

linux图像显示 linux图像显示 (一) framebuffer操作 linux图像显示 (二) bmp图片 linux图像显示 (三) 使用libjpg处理jpg图片 linux图像显示 (四) 使用libp...

freetype安装使用详解(矢量字体操作)

weixin 33994429的博客 **①** 756

freetype下载地址: http://sourceforge.net/projects/freetype/files/ 在PC: tar xjf freetype-2.4.10.tar.bz2 ./configure make sudo make install /* 默认安装在...

qq_41509604: 非常喜欢这种从源码开始讲 解框架的, 赞一个

live555源码分析(八)多播

harbap: 谢谢作者,长篇讲解,现在对live5 55 有了一个基本的了解。

您愿意向朋友推荐"博客详情页"吗?











强烈不推荐 不推荐 一般般 推荐 强烈推荐

深入浅出MySQL事务(二)MVCC的实现原

深入浅出MySQL事务 (一)事务隔离

深入浅出MySQL索引(二)InnoDB存储引擎 的索引

2020年 4篇 2019年 95篇

2018年 7篇

freetype 矢量字体

可以制作汉,英矢量字体,可以调节不同字体大小。

通过freetype库实现矢量显示

本章主要内容如下: 1) **矢量字体**原理 2) 使用freetype库实现**矢量字体显示** 1. **矢量字体**原理 将汉字的笔划边缘用直线段描述成封闭的曲线,并将线段各...

freetype2使用基础程序

自由空间,实现思想的自由之地 ① 1850

#include #include #include // freetype2 头文件#include FT_FREETYPE_H // for freetype support#include FT_GLYPH_H// 最终文本内容存放的...

1.tar xvj freetype-2.4.10.tar.bz2 2 cd freetype-2.4.10/3 ./configure 4 make 5 sudo make install /* /usr/local/lib/*/ 1.如此时编译 gcc show_fontprint.c -o ...

freetype 使用解析---矢量字体 最新发布

weixin 42889383的博客 ① 687

u012996559的博客 ① 1017

06-05

10-12

屏幕<mark>显示字体(字体</mark>点阵): 在内核中有对应的文件fontdata_8x16.c,将字体通过数组8*16来描述,0表示为空,1表示描点,通过通过各个点形成一个字体...

w-hbin 的专栏 ① 4583 FreeType2教程

第一步 - 「简易的字形装载 介绍 这是"FreeType2 教程"的第一部分。它将教会你如何:*初始化库*通过创建一个新的 face 对象来打开一个字体文件*...

stm32矢量字体freetype.rar

Freetype在STM32的移植,使用的freetype版本为2.10,实际项目中可能需要做小的修改

©2021 CSDN 皮肤主题: 数字20 设计师:CSDN官方博客 返回首页

关于我们 招贤纳士 广告服务 开发助手 ☎ 400-660-0108 ☑ kefu@csdn.net ⑤ 在线客服 工作时间 8:30-22:00

및 JT同学 (关注)

















