

原 ffdshow 源代码分析 3：位图覆盖滤镜（设置部分Settings）

2013年10月24日 23:52:21 阅读数：4001

=====

ffdshow源代码分析系列文章列表：

[ffdshow 源代码分析 1：整体结构](#)

[ffdshow 源代码分析 2：位图覆盖滤镜（对话框部分Dialog）](#)

[ffdshow 源代码分析 3：位图覆盖滤镜（设置部分Settings）](#)

[ffdshow 源代码分析 4：位图覆盖滤镜（滤镜部分Filter）](#)

[ffdshow 源代码分析 5：位图覆盖滤镜（总结）](#)

[ffdshow 源代码分析 6：对解码器的dll的封装（libavcodec）](#)

[ffdshow 源代码分析 7：libavcodec视频解码器类（TvideoCodecLibavcodec）](#)

[ffdshow 源代码分析 8：视频解码器类（TvideoCodecDec）](#)

[ffdshow 源代码分析 9：编解码器有关类的总结](#)

=====



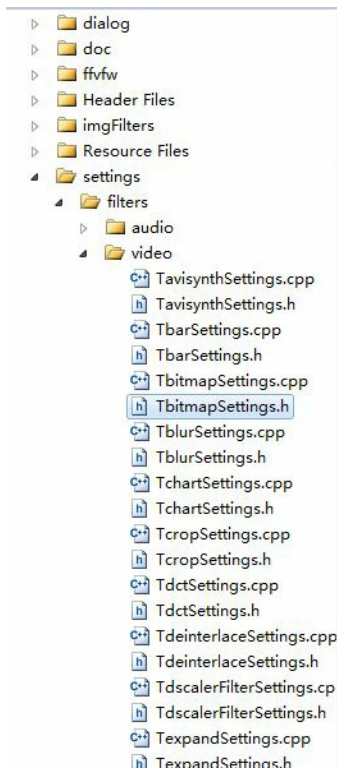
上一篇文章介绍了ffdshow的位图覆盖滤镜的对话框（Dialog）部分：[ffdshow 源代码分析2：位图覆盖滤镜（对话框部分Dialog）](#)

在这里再介绍一下设置部分（Settings），此外还有一个滤镜部分（Filter）。这三个部分就可以组成一个ffdshow的滤镜功能了。

设置部分（Settings）

在ffdshow中滤镜的设置部分（Settings）主要用于存储滤镜运行过程中需要用到的各种变量。一般情况下通过读取注册表变量并赋值给该类当中的变量从而达到操作相应滤镜的功能。

与位图覆盖（Bitmap）滤镜的设置有关的类位于settings->filters->video目录下（隐藏的很深啊）的TbitmapSettings.cpp和TbitmapSettings.h文件中。



先来看看TbitmapSettings.h

该类的名字叫TbitmapSettings，从类的定义我们可以看出，

flnm[]存储了打开的位图的路径

posx, posy存储了位图在屏幕上显示的位置

mode存储了显示的方式

等等，所有跟该滤镜（Filter）相关的数据都存储在该类之中。

该类包含一个TfilterIDFF类型的结构体idffs，用于存储该滤镜的一些属性信息（名称，ID，属性对话框ID等等）

此外，有两个函数至关重要。createFilters()用于创建滤镜（Filter）。createPages()用于创建滤镜的配置对话框（Dialog）。

```

1.  #ifndef _TBITMAPSETTINGS_H_
2.  #define _TBITMAPSETTINGS_H_
3.  //各个Filter预设值
4.  #include "TfilterSettings.h"
5.  #include "Tfont.h"
6.  //Bitmap的配置信息
7.  struct TbitmapSettings : TfilterSettingsVideo {
8.  private:
9.      static const TfilterIDFF idffs;
10. protected:
11.     virtual const int *getResets(unsigned int pageId);
12. public:
13.     TbitmapSettings(TintStrColl *Icoll = NULL, TfilterIDFFs *filters = NULL);
14.     //Bitmap文件路径
15.     char_t flnm[MAX_PATH];
16.     //x,y坐标, 以及坐标的模式
17.     int posx, posy, posmode;
18.     int align;
19.     //叠加方式
20.     enum {
21.         MODE_BLEND = 0,
22.         MODE_DARKEN = 1,
23.         MODE_LIGHTEN = 2,
24.         MODE_ADD = 3,
25.         MODE_SOFTLIGHT = 4,
26.         MODE_EXCLUSION = 5
27.     };
28.     int mode;
29.     static const char_t *modes[];
30.     int strength;
31.     //创建Filter
32.     virtual void createFilters(size_t filtersorder, Tfilters *filters, TfilterQueue &queue) const;
33.     //创建属性页面
34.     virtual void createPages(TffdshowPageDec *parent) const;
35.     virtual bool getTip(unsigned int pageId, char_t *buf, size_t buflen);
36. };
37.
38. #endif

```

再看看 TbitmapSettings.cpp

该类包含了TbitmapSettings类中函数方法的具体实现。首先看一下构造函数TbitmapSettings()。从构造函数中可以看出，绑定了类中的变量和注册表变量，使它们形成一一对应的关系。其他的函数就不再细说了，比较简单，理解起来比较容易。

```

1.  /*
2.   * Copyright (c) 2004-2006 Milan Cutka
3.   *
4.   * This program is free software; you can redistribute it and/or modify
5.   * it under the terms of the GNU General Public License as published by
6.   * the Free Software Foundation; either version 2 of the License, or
7.   * (at your option) any later version.
8.   *
9.   * This program is distributed in the hope that it will be useful,
10.  * but WITHOUT ANY WARRANTY; without even the implied warranty of
11.  * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
12.  * GNU General Public License for more details.
13.  *
14.  * You should have received a copy of the GNU General Public License
15.  * along with this program; if not, write to the Free Software
16.  * Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
17.  */
18.
19. #include "stdafx.h"
20. #include "TbitmapSettings.h"
21. #include "TimgFilterBitmap.h"
22. #include "Cbitmap.h"
23. #include "TffdshowPageDec.h"
24. #include "TsubtitlesSettings.h"
25. //几种叠加方式
26. const char_t* TbitmapSettings::modes[] = {
27.     _L("blend"),
28.     _L("darken"),
29.     _L("lighten"),
30.     _L("add"),
31.     _L("softlight"),
32.     _L("exclusion"),
33.     NULL
34. };
35. //Filter属性
36. const TfilterIDFF TbitmapSettings::idffs = {
37.     /*name*/      _L("Bitmap overlay"),
38.     /*id*/        IDFF_filterBitmap,
39.     /*is*/        IDFF_isBitmap,

```

```

40.     /*order*/      IDFF_orderBitmap,
41.     /*show*/       IDFF_showBitmap,
42.     /*full*/       IDFF_fullBitmap,
43.     /*half*/       0,
44.     /*dlgId*/      IDD_BITMAP,
45. };
46. //构造函数
47. TbitmapSettings::TbitmapSettings(TintStrColl *Icoll, TfilterIDFFs *filters): TfilterSettingsVideo(sizeof(*this), Icoll, filters, &idffs)
48. {
49.     half = 0;
50.     memset(flnm, 0, sizeof(flnm));
51.     //绑定变量
52.     static const TintOptionT<TbitmapSettings> iopts[] = {
53.         IDFF_isBitmap      , &TbitmapSettings::is      , 0, 0, _l(""), 1,
54.         _l("isBitmap"), 0,
55.         IDFF_showBitmap    , &TbitmapSettings::show    , 0, 0, _l(""), 1,
56.         _l("showBitmap"), 1,
57.         IDFF_orderBitmap   , &TbitmapSettings::order   , 1, 1, _l(""), 1,
58.         _l("orderBitmap"), 0,
59.         IDFF_fullBitmap    , &TbitmapSettings::full    , 0, 0, _l(""), 1,
60.         _l("fullBitmap"), 0,
61.         IDFF_bitmapPosx    , &TbitmapSettings::posx    , -4096, 4096, _l(""), 1,
62.         _l("bitmapPosX"), 50,
63.         IDFF_bitmapPosy    , &TbitmapSettings::posy    , -4096, 4096, _l(""), 1,
64.         _l("bitmapPosY"), 50,
65.         IDFF_bitmapPosmode , &TbitmapSettings::posmode , 0, 1, _l(""), 1,
66.         _l("bitmapPosMode"), 0,
67.         IDFF_bitmapAlign   , &TbitmapSettings::align   , 0, 3, _l(""), 1,
68.         _l("bitmapAlign"), ALIGN_CENTER,
69.         IDFF_bitmapMode    , &TbitmapSettings::mode    , 0, 5, _l(""), 1,
70.         _l("bitmapMode"), 0,
71.         IDFF_bitmapStrength , &TbitmapSettings::strength , 0, 256, _l(""), 1,
72.         _l("bitmapStrength"), 128,
73.         0
74.     };
75.     addOptions(iopts);
76.     static const TstrOption sopts[] = {
77.         IDFF_bitmapFlnm    , (TstrVal)&TbitmapSettings::flnm , MAX_PATH, 0, _l(""), 1,
78.         _l("bitmapFlnm"), _l(""),
79.         0
80.     };
81.     addOptions(sopts);
82.
83.     static const TcreateParamList1 listMode(modes);
84.     setParamList(IDFF_bitmapMode, &listMode);
85.     static const TcreateParamList1 listAlign(TsubtitlesSettings::alignments);
86.     setParamList(IDFF_bitmapAlign, &listAlign);
87. }
88. //创建Filter
89. void TbitmapSettings::createFilters(size_t filtersorder, Tfilters *filters, TfilterQueue &queue) const
90. {
91.     idffOnChange(idffs, filters, queue.temporary);
92.     if (is && show) {
93.         queueFilter<TimgFilterBitmap>(filtersorder, filters, queue);
94.     }
95. }
96. //创建属性页面
97. void TbitmapSettings::createPages(TffdshowPageDec *parent) const
98. {
99.     parent->addFilterPage<TbitmapPage>(&idffs);
100. }
101.
102. const int* TbitmapSettings::getResets(unsigned int pageId)
103. {
104.     static const int idResets[] = {
105.         IDFF_bitmapPosx, IDFF_bitmapPosy, IDFF_bitmapPosmode, IDFF_bitmapAlign, IDFF_bitmapMode, IDFF_bitmapStrength,
106.         0
107.     };
108.     return idResets;
109. }
110.
111. bool TbitmapSettings::getTip(unsigned int pageId, char_t *tipS, size_t len)
112. {
113.     if (flnm[0]) {
114.         tsprintf_s(tipS, len, _TRUNCATE, _l("%S %S"), modes[mode], flnm);
115.         tipS[len - 1] = '\0';
116.     } else {
117.         tipS[0] = '\0';
118.     }
119.     return true;
120. }

```

版权声明：本文为博主原创文章，未经博主允许不得转载。 <https://blog.csdn.net/leixiaohua1020/article/details/13004051>

文章标签：[ffdshow](#) [滤镜](#) [位图叠加](#) [directshow](#) [源代码](#)

个人分类：[ffdshow](#)

所属专栏：[开源多媒体项目源代码分析](#)

此PDF由[spygg](#)生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com