Subject: Re: Draw stuff
Posted by Indio on Fri, 01 Feb 2008 22:41:51 GMT
View Forum Message <> Reply to Message

That's it. I have done some selections, only the important parts of the source are here.

MainWindow.h :

```
#include "Segmentation.h"
#include <CtrlLib/CtrlLib.h>

class Segmentation; // forward class declaration

class MainWindow : public TopWindow {

public:
 typedef MainWindow CLASSNAME;
 MainWindow(  );

 void OpenImage(  );

 ImageCtrl inImgctrl;
 ImageCtrl outImgctrl;

 Splitter h, v, v1, v2;
       Button a;
 One < StreamRaster > r;

 Segmentation *segmentation;

};
```

MainWindow.cpp :

```
#include "MainWindow.h"

using namespace Upp;

//---------------------------

MainWindow::MainWindow(  )
{
 segmentation = NULL;
 segmentation = new Segmentation(  );
```

```
 Title("Statistical Segmentation").Zoomable().Sizeable();
 SetRect(0, 0, 900, 800);

 h.Horz(a, v);
 v.Vert(v1, v2 );
 Add(h.SizePos());
 v1.Add(inImgctrl);
}

//----------------------------

void MainWindow::OpenImage(  )
{
  String fileName = "";
  FileSel fs;

  // If user selects a file to open, returns true
  if ( fs.Type( "bitmap", "*.bmp").ExecuteOpen("Choose the image file to open") )
    {
      fileName = ~fs; // fileName contains the file name

      if ( fileName != "" )
      {
 inImgctrl.SetImage( Null ); // maybe not necessary
 FileIn in( fileName );
 r = StreamRaster::OpenAny( in );
 Image img = StreamRaster::LoadFileAny(~fileName);

        if ( !r )
        {
          return; // invalid input
 }

        ...

 if ( segmentation->LoadImage( r ) )
 {
   outImgctrl.SetImage( segmentation->ib );
 }
      }
    }
}
```

Segmentation.h :

```
#include "MainWindow.h"
#include <CtrlLib/CtrlLib.h>

class Segmentation {

public:
 typedef Segmentation CLASSNAME;

 Segmentation(  );
 // Does the evolution
 bool Iteration(  );
 // Called from MainWindow, generates the matrix of the image
 bool LoadImage( One < StreamRaster > streamRaster );

 ImageBuffer ib;

 // contains the intensities of the input image
 int **picture;

protected:
 void Init( One < StreamRaster > streamRaster );
};
```

Segmentation.cpp :

```
Segmentation::Segmentation(  )
{
 imageHeight = -1;
 imageWidth = -1;
 picture = NULL;
}

//---------------------------------

bool Segmentation::LoadImage( One < StreamRaster > r )
{
 imageHeight = r->GetHeight();
 imageWidth = r->GetWidth();

 Init( r );

 if ( (imageHeight != -1) && (imageWidth != -1) )
  return true;
 else
  return false;
}
```

```
//---------------------------------

void Segmentation::Init( One < StreamRaster > r )
{
 // memory allocations
 picture = new int* [imageWidth];
 for ( int i = 0; i < imageWidth; i++ )
  picture[i] = new int[imageHeight]; // the image intensities

 // image -> matrix
 for(int i = 0; i < imageHeight; i++) {
  RasterLine l = r->GetLine(i);
  for(int j = 0; j < imageWidth; j++) {
   picture[i][j] = l[j].b;
  }
 }

        ...

 // matrix -> image

 ib(imageHeight, imageWidth);

 for(int y = 0; y < imageWidth; y++) {
     RGBA *l = ib[y];
     for(int x = 0; x < imageHeight; x++) {
                l->a = 255;
                l->r = picture[y][x];
                l->g = picture[y][x];
                l->b = picture[y][x];
                l++;
     }
 }

        //Premultiply(ib);
}
```

I hope this is enough. Thanks!