

## Re: display array in a frame wnd

---

*Source:* <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2007-09/msg00190.html>

---

- *From:* "J-F Portala" <jfportala@xxxxxxx>
  - *Date:* Wed, 5 Sep 2007 22:20:48 +0200
- 

Thank you Joseph for your help.

"Joseph M. Newcomer" <newcomer@xxxxxxxxxxxx> a écrit dans le message de news: 24otd3t718oaqcts5oi46fnd7fob0eii2i@xxxxxxxxxxx

First, you have to create an app that is suitable for displaying an image.  
There are  
several options:

A CDialog-based app, with a CStatic control in which the image is displayed

An SDI app, where the image is displayed in the view.

In general, I use single document without document view architecture.

An MDI app, where the image is displayed in the view.

The details differ slightly in how you do this, but ultimately, there is an OnPaint handler for the subclassed CStatic in the dialog or an OnDraw handler for the CView-derived class in your SDI or MDI app. All your drawing logic belongs in the OnPaint/OnDraw handler.

There would never be a need to create a frame window; it is not at all clear why such code would be written. Such code is automatically generated by the AppWizard.

But I need several windows (2,3 or 10).  
Some of them can be with special menu and status bar.  
When I move with the mouse on a window, in the status bar,  
I want to show location of the mouse in the image and value of pixel at this point.

There is no reason to do a DestroyWindow, or create a new window, to do

## Re: display array in a frame wnd

the display; the window is already there. An approach which does not make this assumption is probably the wrong approach.

When a window corresponding to a image result is no more usefull, I think it is better to desestroy it.

In an SDI app or MDI app there is already a status bar, menus, and toolbars. There is an MSDN article on how to add toolbars and status bars to a dialog-based app, but for this project, it strikes me that an SDI or MDI app would be the correct choice. Since all the "future enhancements" you think you want are already present, and cost you zero effort, there is no reason to take some macho I-have-to-write-everything-myself approach.

I did not have enough time free to do that, that is the reason why I want to be sure to make the good choice.

I better prefered to reuse display provided with the librairy opencv, but I can obtain handle of the window.

but when I attach it to a CWnd (with FromHandle), the window does not react to ShowWindow(SW\_SHOW) or CreateStatusBar.

```
cvNamedWindow("test",0) ; // window is created
cvShowImage("test",pImg) ; // window is displayed with the array pImg
// just two lines to create and display the image
cvSetMouseCallBack("test",on_mouse,this) ; // permits to define a static
function on_mouse where it is possible to catch mouse events
```

The creation of window does not use MFC, but I was thinking it was possible to create a MFC window starting from HWND

```
// the following is what I have tried
hWnd = (HWND) cvGetWindowHandle("test") ; // handle of the window
pWnd = FromHandle(hWnd) ;// attach to a CWnd
pStatusBar->Create(pWnd) ;
```

but when I want to create a CStatusBar, it displays nothing in the window. pWnd->ShowWindow(SW\_HIDE) does not work too.

So I can't attach a menu too.

Such an

attitude is counterproductive, and not even sensible. I think the fundamental failure you have made here is in not using the correct tools to create an app, instead

## Re: display array in a frame wnd

thinking you  
have a chance of doing this from scratch. Such an approach is usually  
doomed; very few  
MFC programmers would adopt such an approach, and the ones who do are  
usually very  
sophisticated MFC programmers. This is *\*never\** a viable approach for a  
beginner, no  
matter what bizarre book you are using that suggests it could possibly  
make sense. It is  
just an inappropriate approach, and should be instantly scrapped.

I think I have a big lack in how displaying a bitmap. I have never worked  
with document view architecture,  
because I think it is not adapted to my applications.  
I always created projects in Single Document without document/view  
architecture.

So start over. Create either an MDI or SDI app, copy code across, make  
sure ALL drawing  
is done in the OnDraw handler of the CView-derived class. There is no  
need to create new  
windows, so the approach that requires creating and destroying windows  
should be  
abandoned.  
joe