

کتابخانه‌ی اصلی OpenCV، دارای دو نوع اینترفیس C و C++ است. اینترفیس C آن مرتبط است به نگارش‌های 1x آن و اینترفیس C++ آن به همراه نگارش‌های 2x آن ارائه شده‌اند. کتابخانه‌ی OpenCVSharp هر دو نوع اینترفیس یاد شده را پشتیبانی می‌کند. در این قسمت نگاهی خواهیم داشت به نحوه‌ی بارگذاری و نمایش تصاویر در OpenCV به کمک متدهای اینترفیس C آن، مانند `.cvLoadImage`، `cvShowImage`، `cvReleaseImage`.

بارگذاری و نمایش تصاویر به کمک OpenCVSharp

متدهای اینترفیس C مربوط به OpenCV، در OpenCVSharp با ذکر کلاس Cv آن قابل دسترسی هستند. برای نمونه متدهای C یاد شده‌ی در ابتدای بحث، چنین معادلی را در OpenCVSharp دارند:

```
using OpenCvSharp;

namespace OpenCvSharpSample02
{
    class Program
    {
        static void Main(string[] args)
        {
            var img = Cv.LoadImage(@"..\..\images\ocv02.jpg");

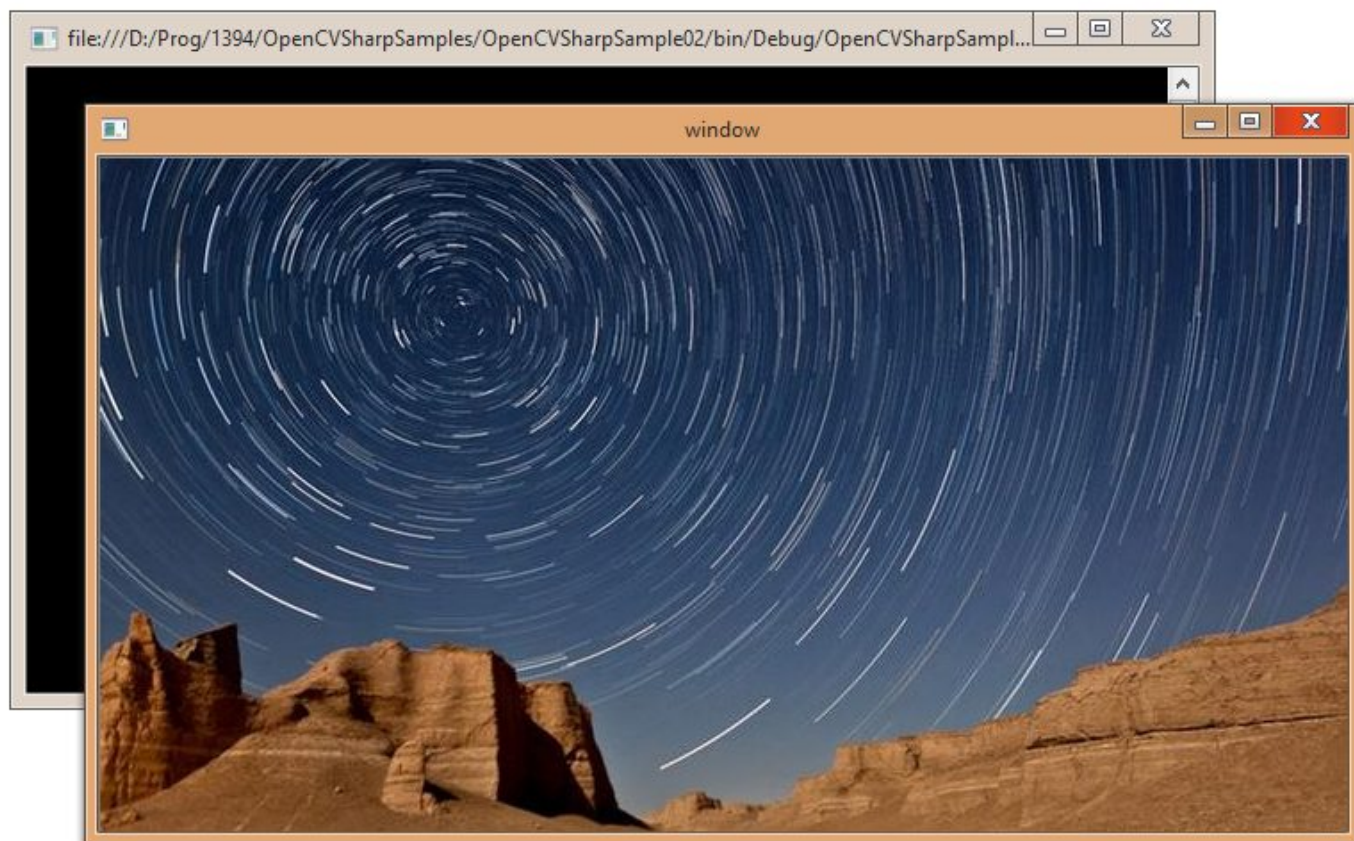
            Cv.NamedWindow("window");
            Cv.ShowImage("window", img);

            Cv.WaitKey();

            Cv.DestroyWindow("window");

            Cv.ReleaseImage(img);
        }
    }
}
```

متد `cvLoadImage` اینترفیس C، به `Cv.LoadImage` تبدیل شده‌است و مابقی نیز به همین ترتیب. در اینجا با استفاده از متد `LoadImage`، تصویری را از مسیر مشخصی، بارگذاری می‌کنیم. سپس یک پنجره‌ی OpenCV ایجاد و این تصویر در آن نمایش داده می‌شود. متد `WaitKey` منتظر فشرده شدن یک کلید بر روی پنجره‌ی OpenCV می‌شود. پس از آن این پنجره تخریب و همچنین منابع native این تصویر آزاد می‌شوند.



متد LoadImage، پارامتر دومی را نیز می‌پذیرد:

```
var img = Cv.LoadImage(@"..\..\images\ocv02.jpg", LoadMode.GrayScale);
```

برای مثال در اینجا می‌توان به کمک مقدار LoadMode.GrayScale، تصویر را به صورت سیاه و سفید بارگذاری کرد. Enum تعریف شده‌ی در اینجا قابلیت or یا جمع منطقی را نیز دارد. برای مثال می‌توان مقدار LoadMode.AnyColor | LoadMode.AnyDepth را نیز مشخص کرد؛ جهت بارگذاری تصویر اصلی با مشخصات کامل آن که حالت پیش فرض است.

کلاس‌های پشت صحنه‌ی اینترفیس C در OpenCVSharp

علت وجود کلاس Cv در OpenCVSharp، سهولت برگرداندن مثال‌های C کتابخانه‌ی OpenCV به نمونه‌های دات نتی است. اما اگر قصد داشته باشید از کلاس‌های پشت صحنه‌ی این اینترفیس در OpenCVSharp استفاده کنید، می‌توان کدهای فوق را به نحو ذیل نیز بازنویسی کرد:

```
using (var img = new IplImage(@"..\..\images\ocv02.jpg", LoadMode.Unchanged))
{
    using (var window = new CvWindow("window"))
    {
        window.Image = img;
        Cv.WaitKey();
    }
}
```

خروجی متد LoadImage از نوع کلاس IplImage است. در اینجا می‌توان همین کلاس را وهله سازی کرد و مورد استفاده قرار داد. به علاوه اینبار این کلاس تهیه شده، اینترفیس IDisposable را نیز پیاده سازی می‌کند. بنابراین می‌توان با استفاده از عبارت using کار آزاد سازی منابع آن را خودکار کرد.

همچنین پنجره‌ی OpenCV نیز در اینجا با کلاس CvWindow پیاده سازی می‌شود که این کلاس نیز اینترفیس IDisposable را پیاده سازی می‌کند.

یک نکته‌ی تکمیلی

اگر متد LoadImage کتابخانه‌ی OpenCV قادر به بارگذاری تصویر شما نبود، متد دیگری به نام IplImage.FromFile نیز پیش بینی شده‌است. این متد از امکانات System.Drawing.Bitmap دات نت برای بارگذاری تصویر و تبدیل آن به فرمت OpenCV استفاده می‌کند.

کدهای کامل این مثال را [از اینجا](#) می‌توانید دریافت کنید.