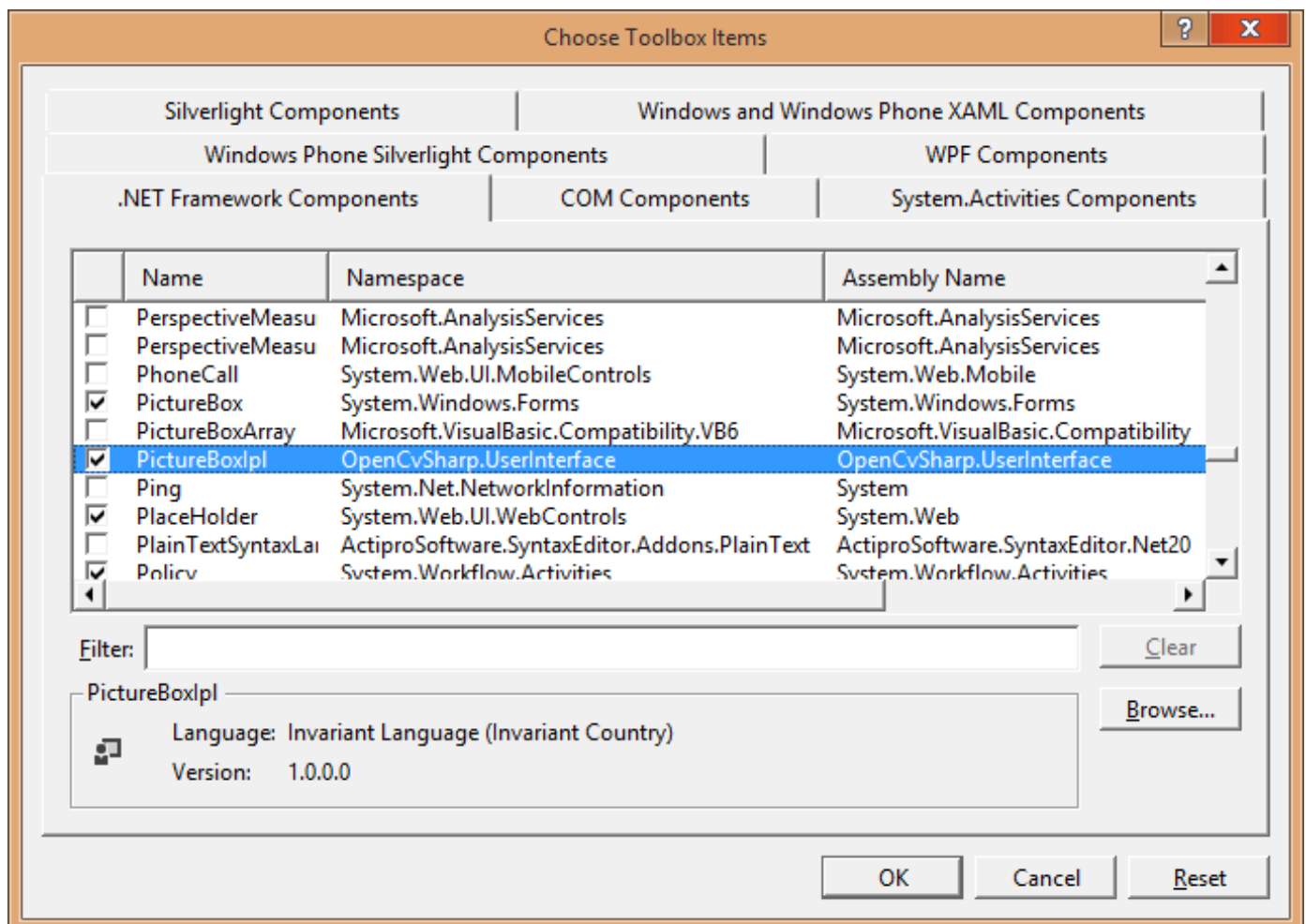


استفاده از پنجره‌ی native خود OpenCV، روش مرسوم‌ی است در زبان‌های مختلف برنامه نویسی که از OpenCV استفاده می‌کنند و این پنجره مستقل است از سکوی کاری مورد استفاده. اما شاید در دات نت علاقمند باشید که نتیجه‌ی عملیات را در یک picture box استاندارد نمایش دهید. در ادامه، تبدیل تصاویر OpenCV را به فرمت دات نت، در دو قالب برنامه‌های WinForms و همچنین WPF، بررسی خواهیم کرد.

استفاده از OpenCVSharp در برنامه‌های WinForms به کمک PictureBoxIpl

یکی از اسمبلی‌های کتابخانه‌ی OpenCVSharp را که در پوشه‌ی bin برنامه می‌توان مشاهده کرد، OpenCvSharp.UserInterface.dll نام دارد. این اسمبلی حاوی یک picture box جدید به نام PictureBoxIpl است که می‌تواند تصاویری را با فرمت IplImage، دریافت کند.



می‌توانید این picture box ویژه را از طریق منوی `ToolBox -> Choose items` و سپس صفحه‌ی دیالوگ فوق، به نوار ابزار WinForms اضافه کرده و از آن استفاده کنید و یا می‌توان با کدنویسی نیز به آن دسترسی یافت:

```
using (var iplImage = new IplImage(@"..\..\Images\Penguin.png", LoadMode.Color))
```

```
{
    Cv.Dilate(iplImage, iplImage);

    var pictureBoxIpl = new OpenCvSharp.UserInterface.PictureBoxIpl
    {
        ImageIpl = iplImage,
        AutoSize = true
    };
    flowLayoutPanel1.Controls.Add(pictureBoxIpl);
}
```

در اینجا تصویر مورد نظر را توسط کلاس IplImage بارگذاری کرده و سپس برای نمونه فیلتر Dilate را به آن اعمال کرده‌ایم. سپس وهله‌ی جدیدی از کنترل PictureBoxIpl ایجاد و خاصیت ImageIpl آن، به تصویر بارگذاری شده، تنظیم و در آخر این picture box با کدنویسی به صفحه اضافه شده‌است.

یک نکته

هر نوع تغییری به iplImage پس از انتساب آن به خاصیت ImageIpl، نمایش داده نخواهد شد. برای به حداقل رساندن سربار ایجاد اشیاء جدید (خصوصاً برای نمایش اطلاعات رسیده‌ی از دوربین یا WebCam)، از متد RefreshIplImage استفاده کنید. این متد بجای ایجاد یک شیء جدید، تنها ناحیه‌ی موجود را مجدداً ترسیم خواهد کرد و بسیار سریع است:

```
pictureBoxIpl.RefreshIplImage(iplImage);
```

استفاده از OpenCVSharp در برنامه‌های WinForms به کمک PictureBox

اگر نخواهید از کنترل جدید PictureBoxIpl استفاده کنید، می‌توان از همان Picture box استاندارد WinForms نیز کمک گرفت:

```
Bitmap bitmap;
using (var iplImage = new IplImage(@"..\..\Images\Penguin.png", LoadMode.Color))
{
    bitmap = iplImage.ToBitmap(); // BitmapConverter.ToBitmap()
}

var pictureBox = new PictureBox
{
    Image = bitmap,
    ClientSize = bitmap.Size
};

flowLayoutPanel1.Controls.Add(pictureBox);
```

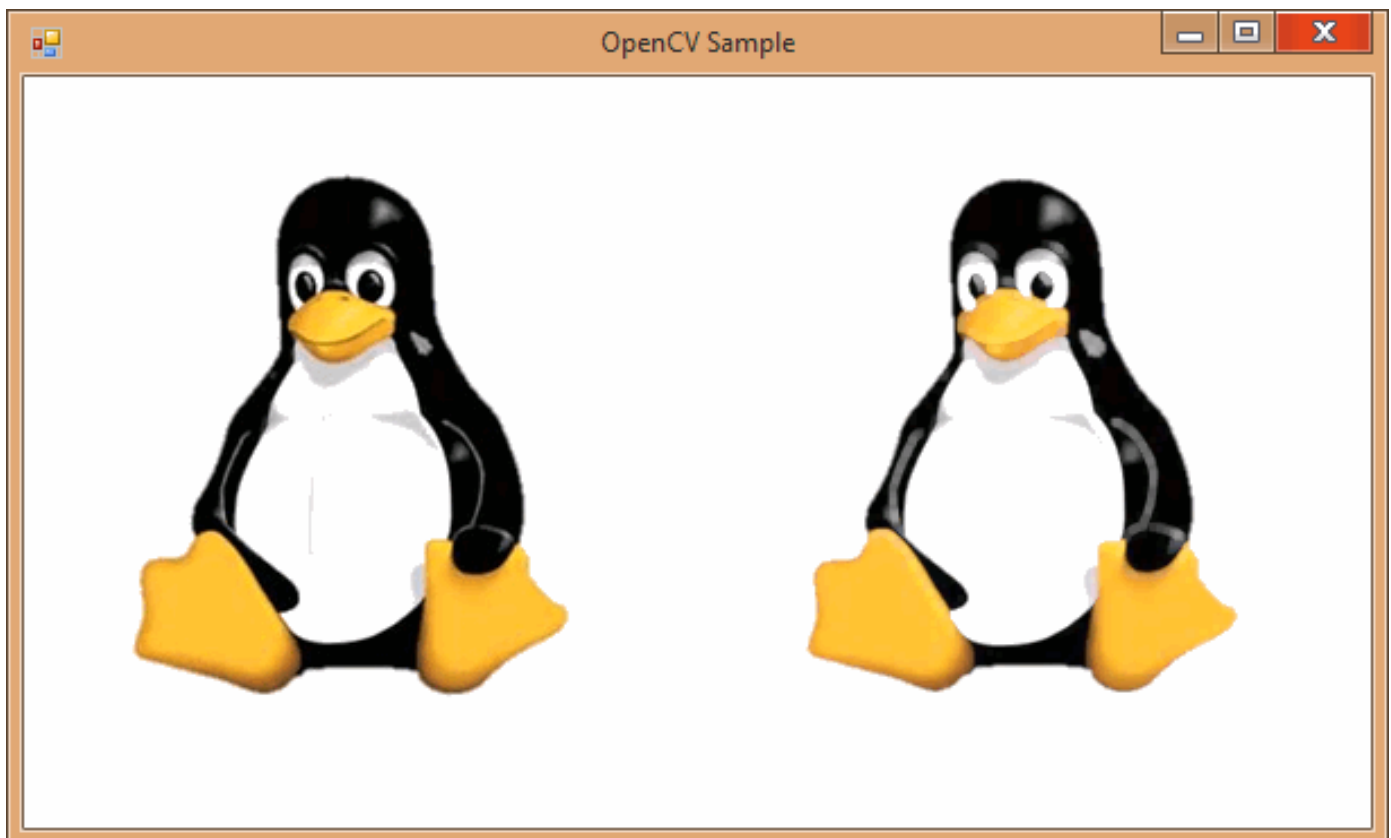
تنها نکته‌ای که در اینجا جدید است، استفاده از متد الحاقی ToBitmap می‌باشد که در کلاس BitmapConverter کتابخانه‌ی OpenCVSharp تعریف شده‌است. به این ترتیب تصویر با فرمت OpenCV، به یک Bitmap دات نت تبدیل می‌شود. اکنون می‌توان این بیت‌مپ را برای مثال به یک Picture box استاندارد انتساب داد و یا حتی متد Save آن‌را فراخوانی کرد و آن‌را بر روی دیسک سخت، ذخیره نمود.

یک نکته

در اینجا نیز برای به حداقل رسانی به روز رسانی‌های بعدی picture box بهتر است از متد ToBitmap به شکل زیر کمک گرفت:

```
iplImage.ToBitmap(dst: (Bitmap)pictureBox.Image);
```

به این ترتیب سربار وهله سازی یک شیء جدید Bitmap حذف خواهد شد و صرفاً ناحیه‌ی نمایشی مجدداً ترسیم می‌شود.



استفاده از OpenCVSharp در برنامه‌های WPF

در WPF می‌توان با استفاده از متد الحاقی `ToWriteableBitmap` کلاس `BitmapConverter`، فرمت `IplImage` را به منبع تصویر یک کنترل تصویر استاندارد، تبدیل کرد:

```
using System.Windows.Media;
using OpenCvSharp;
using OpenCvSharp.Extensions;

namespace OpenCVSharpSample05Wpf
{
    public partial class MainWindow
    {
        public MainWindow()
        {
            InitializeComponent();
            loadImage();
        }

        private void loadImage()
        {
            using (var iplImage = new IplImage(@"..\..\Images\Penguin.png", LoadMode.Color))
            {
                Cv.Dilate(iplImage, iplImage);

                Image1.Source = iplImage.ToWriteableBitmap(PixelFormats.Bgr24);
            }
        }
    }
}
```

کدهای کامل [WPF](#) و [WinForms](#) این مطلب برای دریافت.

نظرات خوانندگان

نویسنده: علی ساری
تاریخ: ۱۳۹۴/۰۳/۱۶ ۸:۳۰

سلام؛ ممنون از مطلب خوبتون. من از کتابخانه‌های opencv و emgu برای خوندن تصویر از وب کم و پردازش تصویر استفاده میکنم. مشکل من سرعت پایین برنامه در رزولوشن بالای وب کم است هر چه رزولوشن بالاتر میره سرعت برنامه من کمتر میشه. مثلا تو یه سیستم corei7 با ram 8 gig من 61 درصد استفاده از cpu دارم و فیلم نمایش داده شده در برنامه از محیط واقعی عقب‌تر است. من در رویداد Application.Idle فرم این کد را قرار دادم:

```
public void Application_Idle(object sender, EventArgs e)
{
    if (_capture != null)
    {
        try
        {
            frame = _capture.QueryFrame();
            Pic.Image = frame.ToBitmap();
            //frame.ToBitmap(dst: (Bitmap)Pic.Image);
        }
        catch (NullReferenceException excpt)
        {
            MessageBox.Show(excpt.Message); // you can also show any suitable message
        }
    }
}
```

شما فرمودید برای به حداقل رسانی به روز رسانی‌های بعدی picture box بهتر است از متد ToBitmap به شکل زیر کمک گرفت:

```
iplImage.ToBitmap(dst: (Bitmap)pictureBox.Image);
```

ولی dst رو فرم نمیشناسه و اینکه آیا رویداد idle رویداد مناسبی برای این کار هست؟ نظرتون درباره سرعت پایین برنامه من چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۴/۰۳/۱۶ ۱۰:۴۷

« [نمایش ویدیو و اعمال فیلتر بر روی آن](#) »