
Subject: Re: Is 'texture mapping' possible on Painter?

Posted by [mirek](#) on Tue, 04 Feb 2020 12:28:10 GMT

[View Forum Message](#) <> [Reply to Message](#)

```
#include <CtrlLib/CtrlLib.h>
#include <plugin/jpg/jpg.h>

using namespace Upp;

Xform2D MapTriangle(Pointf s1, Pointf s2, Pointf s3)
{ // maps 0,0 -> s3, 1,0 -> s1, 0,1 -> s2
    Xform2D s;
    s.x.x = s1.x - s3.x;
    s.y.x = s1.y - s3.y;
    s.x.y = s2.x - s3.x;
    s.y.y = s2.y - s3.y;
    s.t = s3;
    return s;
}

Xform2D MapTriangles(Pointf s1, Pointf s2, Pointf s3, Pointf t1, Pointf t2, Pointf t3)
{
    return Inverse(MapTriangle(s1, s2, s3)) * MapTriangle(t1, t2, t3);
}

struct MyApp : TopWindow {
    Image img;

    virtual void Paint(Draw& dw)
    {
        Size sz = GetSize();
        DrawPainter w(dw, sz);
        w.Co();
        w.DrawRect(sz, White());
        Pointf center = (Point)sz / 2;
        double radius = 0.95 * min(center.x, center.y);

        Size isz = img.GetSize();

        int steps = 200;
        for(double i = 0; i < steps; i++) {
            Pointf q = Polar(i * M_2PI / steps);
            Pointf p1 = radius * q + center;
            Pointf p2 = radius * Polar((i + 1) * M_2PI / steps + 0.4 / steps) + center;

            Xform2D m = Xform2D::Scale(Distance(p1, p2) * steps / isz.cx, radius / isz.cy);
            m = m * Xform2D::Rotation(Bearing(p2 - p1));
        }
    }
}
```

```

Pointf h = p1 - i * isz.cx / steps * Orthogonal(q);
m = m * Xform2D::Translation(h.x, h.y);

double x1 = i * isz.cx / steps;
double x2 = (i + 1) * isz.cx / steps;
double xc = (x1 + x2) / 2;
w.Move(center).Line(p1).Line(p2).Fill(img,
    MapTriangles(Pointf(xc, isz.cy), Pointf(x1, 0), Pointf(x2, 0),
        center, p1, p2));
}

w.Finish();
}
};

GUI_APP_MAIN
{
    MyApp app;
    app.img = StreamRaster::LoadFileAny("C:/xxx/aukro.jpg");
    app.Run();
}

```

(Adding MapTriangles as Xform2D static method...)
