Subject: Is 'texture mapping' possible on Painter?
Posted by Tom1 on Tue, 28 Jan 2020 14:32:32 GMT
View Forum Message <> Reply to Message

Hi,

I would like to 'texture map' an image in a shape of a sector or fan. Is this possible in Painter, like it is in OpenGL using a triangle fan?

(I.e. by defining the center, radius and starting and ending angles of a sector and then fill the area with an Image having its top edge touching the center point and its lower edge touching the arc defined by the radius, stretching out between the starting and ending angles.)

I would like to avoid using OpenGL in this project, partly because this is "just a simple" 2D mapping. However, rendering speed may be a factor here as the data is updated quite frequently.

Best regards,

Tom

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by mirek on Fri, 31 Jan 2020 16:30:09 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Tue, 28 January 2020 15:32Hi,

I would like to 'texture map' an image in a shape of a sector or fan. Is this possible in Painter, like it is in OpenGL using a triangle fan?

(I.e. by defining the center, radius and starting and ending angles of a sector and then fill the area with an Image having its top edge touching the center point and its lower edge touching the arc defined by the radius, stretching out between the starting and ending angles.)

I would like to avoid using OpenGL in this project, partly because this is "just a simple" 2D mapping. However, rendering speed may be a factor here as the data is updated quite frequently.

Best regards,

Tom

Now that is an interesting task... :) I think individual triagles are still affine transformations, so IMO should be possible to do this.

EDIT:

Well, not affine after all, but IMO neither this it possible in OpenGL with simple texture mapping. IMO you need to add some advanced non-affine shaders to do this 100% correctly... Probably

easier to do without the fan at that point, just pixel shader that maps the texture.

Mirek

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by Tom1 on Mon, 03 Feb 2020 09:33:35 GMT
View Forum Message <> Reply to Message

Hi Mirek,

Thanks for the suggestion, but I would really like to do this with Painter only. (Unfortunately, my brain does not have an OpenGL shaders -module installed. ;))

As the sectors in the fan are really less than one degree wide each, it would not hurt to visualize them as a set of narrow triangular wedges, each showing one pixel wide stripe from the original image. So the pixels in the image would be represented as parallel lines with their length increasing while getting further from the center.

Is there a function inside the Painter that can render an Image in a way that each one of the Image corners will land in independently specified coordinates and everything in between simply gets interpolated evenly therein?

Best regards,

Tom

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by mirek on Tue, 04 Feb 2020 10:33:35 GMT
View Forum Message <> Reply to Message

You have to compute the transformation matrix correctly.

Mirek

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...)

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by Tom1 on Tue, 04 Feb 2020 15:11:19 GMT
View Forum Message <> Reply to Message

Hi Mirek!

Absolutely perfect!!! This is an incredibly elegant solution and runs so fast!

Thank you so very much! :)

Best regards,

Tom

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by Didier on Wed, 05 Feb 2020 20:18:35 GMT
View Forum Message <> Reply to Message

Hi Tom,

Could it be possible for you to post a sample image of what the result looks like: original image and the mapped image

[edit] Forget about it, I'm compiling the sample :)

Thank you

Subject: Re: Is 'texture mapping' possible on Painter?
Posted by Didier on Thu, 06 Feb 2020 19:40:40 GMT
View Forum Message <> Reply to Message

This sample code is very interesting,

I never took time to look at what Xform2D was used for (I never needed it) and when I see this example, it looks very powerful and easy to use

Maybe it could be included in the examples