Hi Koldo,

Small fonts always look ugly when they're anti-aliased. I'd suggest to round the values for the painter variant as well, it can make a huge difference:

This preview was generated using this code:void TextTest(Painter& sw) {
 sw.DrawText(10,100, "Hello world! // DrawText()", Roman(12), Black());
 for(int i=0; i<=5; ++i)
  sw.Text(10+0.2*i, 120+i*20, Format("Hello world! // Text() shifted %2vf pixel left",i*0.2),
Roman(12))
    .Fill(Black());
 for(int i=0; i<=5; ++i)
  sw.Text(300, 120+i*20.2, Format("Hello world! // Text() shifted %2vf pixel down",i*0.2),
Roman(12))
    .Fill(Black());
}

Also, you're pictures seem like there are errors in the letter spacing. Not sure where that comes from, I couldn't reproduce it on my system... I think you can safely use DrawText() everywhere, or do some compromise, e.g.void DrawText(Painter &w, double x, double y, int angle, const String &text, Font font, Color color) {
 if(font.GetHeight()>15) {
  w.Begin();
  w.Translate(x, y).Rotate(-angle*M_PI/1800.);
  w.Text(0, 0, text, font).Fill(color);
  w.End();
 } else
  w.DrawText(fround(x), fround(y), angle, text, font, color);
}

Best regards,
Honza

EDIT: Corrected image and code

File Attachments
1) text.png, downloaded 723 times