Archived:Overlaying text and image using QImage

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Qt Quick should be used for all UI development on mobile devices. The approach described in this article (using C++ for the Qt app UI) is deprecated.

Description

Using `QPainter` and `QImage` we can overlay any text/image on another image. The source image(s) can be read from any JPG file and the result image can be stored in another JPG file.

Solution

Header & Library:

```
#include <QPainter>
#include <QImage>
#include <QDateTime>
```

Code:

```
void ImageComposition::createImageWithOverlay()
{
      // Input images (from resources)
      QImage baseImage(":/images/base.jpg");
      QImage overlayImage(":/images/overlay.jpg");

      //Allocate Memory which should be least equal to baseImage
      QImage imageWithOverlay = QImage(baseImage.size(), QImage::Format_ARGB32_Premultiplied);

      QPainter painter(&imageWithOverlay);
      painter.setCompositionMode(QPainter::CompositionMode_Source);
      painter.fillRect(imageWithOverlay.rect(), Qt::transparent);

      // define coordinates
      const int px_baseImage=0, py_baseImage = 0;
      const int px_overlayImage = 500;
      const int py_overlayImage = 0;
      painter.setCompositionMode(QPainter::CompositionMode_SourceOver);

      // draw baseImage on an empty image
      '''painter.drawImage(px_baseImage, py_baseImage, baseImage);'''

      // draw the overlay image
      painter.setCompositionMode(QPainter::CompositionMode_SourceOver);
      '''painter.drawImage(px_overlayImage, py_overlayImage, overlayImage);'''

      // draw the text: as an example its taken as Current Date-Time
      const int px_text = 20;
      const int py_text = 20;

      QDateTime dateTime = QDateTime::currentDateTime();
      QString dateTimeString = dateTime.toString();
      '''painter.drawText(px_text, py_text, dateTimeString);'''

      // save the result image
      bool errorCode = imageWithOverlay.save("C:\\Data\\result.jpg");

      painter.end();
 }
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.