Clock Face Detection – Description

This document outlines the challenge approached with the enclosed software and the algorithm that was developed to solve it.

The Challenge

As the title suggests, this piece of software deals with the detection of analogue clock faces in a variety of pictures, as wide as possible. This problem allows for the use of different techniques, which only combined lead to proper results. It is for example not enough to be able to detect circles in relatively clear images, as clocks do not have to be circular and not every circle has to be a clock. Further, pictures of clocks, especially photographs, are seldom clear enough for objects to be separated and extracted right away. For these reasons, it was decided not to detect the shape of the clock or its face itself but rather to rely on another feature that is common to almost all clocks: Twelve hourly marks or numerals, evenly arranged along on the circumference of a circle. This is in fact one of the most characterizing quality of many clocks. So much so, that even twelve dots situated in a circle can easily remind one of a clock.

Before explaining the algorithm, it should be noted that choosing the twelve marks on the clock face as detection criterion inevitably means, that clocks without them will not be detected. One could argue that the hands of clocks are thus a better signature. However, detecting these in a noisy picture is almost impossible, as their general shape is hardly unique. For a more extensive project, a combination of detecting these and other features might be the best solution. This however, goes beyond the scope of this exercise.

The Algorithm

Pre-processing

Before the algorithm can actually start looking for possible clock faces, the input picture (figure 1) must first be pre-processed. First, the bitmap is converted into a single greyscale channel, as colour and saturation are largely irrelevant for the algorithm. After that, small median and averaging filters are applied to reduce noise in the picture (figure 2). From this image, the length of the gradient in every pixel is calculated, using the X- and Y-derivatives. This gradient image is slightly dilated, to increase edge visibility, concluding the pre-processing (figure 3).

Figure 1 – Input Image

Figure 2 – Greyscale, reduced noise

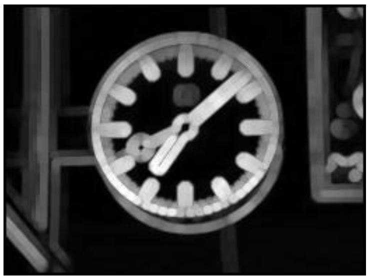

Figure 3 – Dilated Gradient

Finding Clock Faces

Looking at figure 3, the dilated gradient, it becomes clear, why the algorithm does not use the greyscale image to detect clock faces. Not only is the gradient of much higher contrast, but where a clock face can have any brightness, the gradient image specifically highlights high contrast areas without prejudice regarding their grey value. Now, by design, the hourly marks on clocks usually have a high contrast compared to the clock face itself, making them stand out after this operation.

To find possible clock faces from this image, the algorithm now uses a Hough transform. The procedure is carried out over a three dimensional parameter space, spanned by the coordinates of the centre of the clock face and its radius. The feature that is identified specifically are the twelve bright spots caused by the marks or numerals, and the twelve areas in between, which have a much lower value. To achieve this a structuring element, mainly consisting of 24 points with alternating sign, situated in a circle, is used (figure 4).

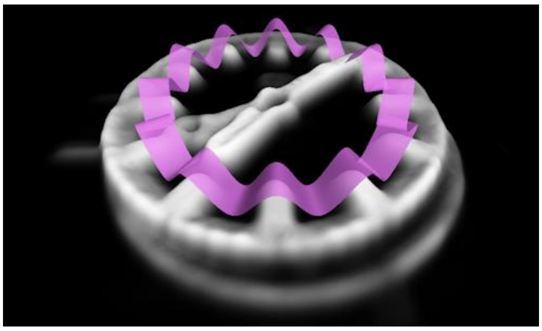

Figure 4 – 3D representation of 'Clock Face' structuring-element fit to gradient image (figure 3)

The result of applying this structure element for various radii to the gradient image is the mentioned parameter space (figure 5). While maxima in this space are often very clear to the human eye, it also contains a lot of noise. Specifically due to the circular nature of the structuring element and the clocks themselves, patterns emerge which are not unlike those caused by wave interference. This means, that just like with interfering waves, there are usually many local extrema. After again reducing noise over the space by averaging over a small volume, interference is filtered out by thresholding the entire space by values carefully chosen based on the overall maximum. The resulting binary space then contains ideally one connected object per clock face (figure 6).

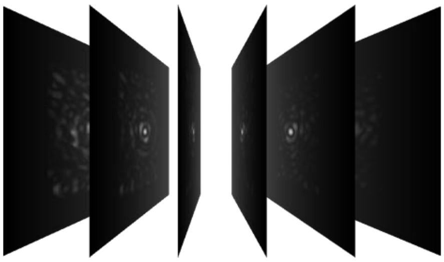

Figure 5 – Parameter Space with noise

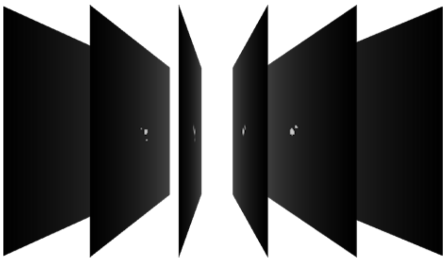

Figure 6 – Thresholded Parameter Space

These objects can now easily be extracted from the parameter space, with their centres of mass corresponding to the coordinates and radii of the clock faces (figure 7).

For more information of the precise details and implementation of the algorithm, please refer to the enclosed source code.

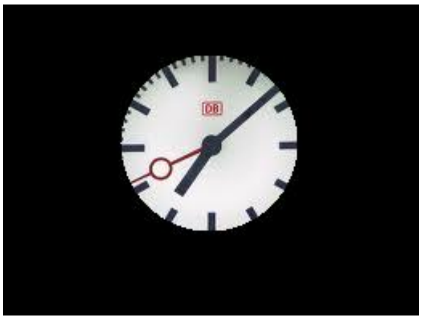

Figure 7 – Detected Clock Face