\subsection(Object Detection)

The general approach used for object detection is a sliding window. The structure of object detection goes as follows:

\begin{itemize}

\item Specific pre-processing for object detection

\item Sliding window: Generation of an error-image (difference from template to reference)

\item Get position of minimum error/maximum correlation

\item Ranking and selecting candidates

\item Return found elements

\subsubsection(Pre-Processing for Object Detection)

Before the sliding window can be applied, the image need to undergo specific Pre-Processing, which is an integral part of object detection. The template (i.e. a resistor) and the reference image (i.e. the circuit) are first binarized and then dilated. The dilating factor is a key-parameter and influences how much tolerance the algorithm allows for dissimilarities between the template image and actual representations of the template in the reference image. Increasing the dilation factor, increases the change to find high-correlation areas, while decreasing accuracy and precision. In figure \ref{detPP1} and figure \ref{detPP2} we can see a template and a reference image being (heavily) dilated.

[insert image detPP1, Caption: Dilated template]

[insert image detPP2, Caption: Dilated reference image]

\subsubsection(Sliding window and error-image)

The pre-processed template is now sliding over the reference image, while a normalized mean-squared error, or difference value, is calculated at each position. The error-values form the error image. In figure \ref{detSW1} and figure \ref{detSW2} you can see the dilated reference image on the left and the error-image on the right. While in figure \ref{detSW1} the dilation value is low, in \ref{detSW2} it is quite high. You can immediately see, that the higher dilated image gets better results, but also more maxima where none should be.

[insert image detSW1, Caption: Slightly dilated reference image and resulting error-image]

[insert image detSW2, Caption: Heavily dilated reference image and resulting error-image]

\subsubsection(Get Maximum Correlation)

Using the error-image (or inverted: the correlation image), the algorithm tries to find significant maxima in the image. Noise and especially similarities between objects (e.g. grounds and resistors, or capacitors and wires) make the search for maxima difficult. The algorithm first supresses all values below a certain dynamic threshold and then binarizes again, using another dynamic threshold. It often happens, that the final binarization fails to produce good results and higher dilation values must be used. Figure \ref{detMX1} to \ref{detMX3} show the different attempts to identify a capacitor (due to its very simple geometric shape, it is notoriously hard to detect with certainty). In figure \ref{detMX1}, we see, that the target maximum is not significant enough and its value is to close to the rest of the error-image. The final binarization results in an all positive image. In figure \ref{detMX2} we see that other areas have significant, although faulty, maxima as well, resulting in a faulty final binarization. In figure \ref{detMX3}, we see a successful minima-suppression and binarization, with an isolated, significant dot. The algorithm detects those isolated dots and decides if it has found something, if it needs to use different parameters or if there is no element of this kind to be found.

[insert image detMX1, Caption: Faulty binarization - Main maximum to insignificant]

[insert image detMX2, Caption: Faulty binarization – False maxima detected]

[insert image detMX3, Caption: Successful binarization]

\subsubsection(Ranking and selecting Candidates)

As mentioned before, the algorithm is prone to confuse different elements with each other, when they share geometric similarity. When dilating the template and reference image, this confusion gets worse, as the tolerance for non-perfect matches increases. It often happens, that the multiple elements are detected (with varying tolerance) in the same area in the reference image. To counteract this, the algorithm assigns a score to each found element relating to its certainty. After the algorithm searched for all elements, it verifies the found elements to be not out of bounds (the edges of the reference image are prone for faulty maxima) and compares their store against each other. In figure \ref{detRS1} you can see all found candidates for this particular example. Attached to each found element you see a score, where lower scores are better. When two or more elements occupy the same area, the element with the lower score gets priority and the others are deleted. In figure \ref{detRS2} you can see the result of this selection process.

[insert image detRS1, Caption: All candidate elements with scores]

[insert image detRS2, Caption: Selected candidates (final result)]

\subsubsection(Return found elements)

The found and selected elements are stored and returned to the main image analysis function for further processing.