# Präsentation Bauteilerkennung

Contents

[Präsentation Projekt 1](#_Toc523462000)

[Einleitung 2](#_Toc523462001)

[Vorlage erstellen 2](#_Toc523462002)

[Bauteile erkennen 7](#_Toc523462003)

[Vorlage Bauteile mousover 11](#_Toc523462004)

[Schluss 11](#_Toc523462005)

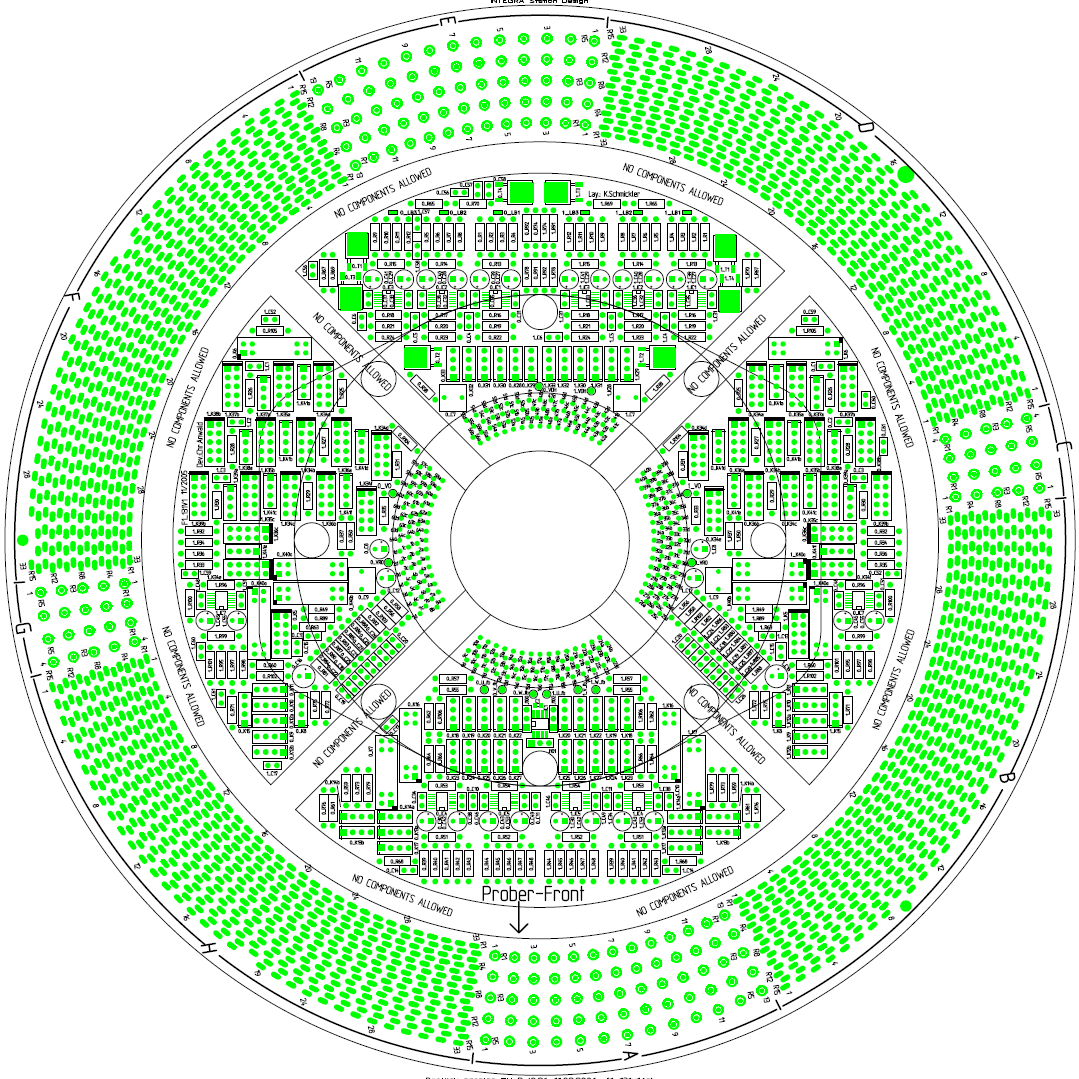
## Einleitung

Ziel dieses Projektes war/ist es, dass ein Foto einer Nadelkarte gemacht wird und durch dieses bestimmt werden kann, ob Bauteile fehlen und wenn welche fehlen, um welches Bauteil es sich handelt. Der grobe Ablauf ist:

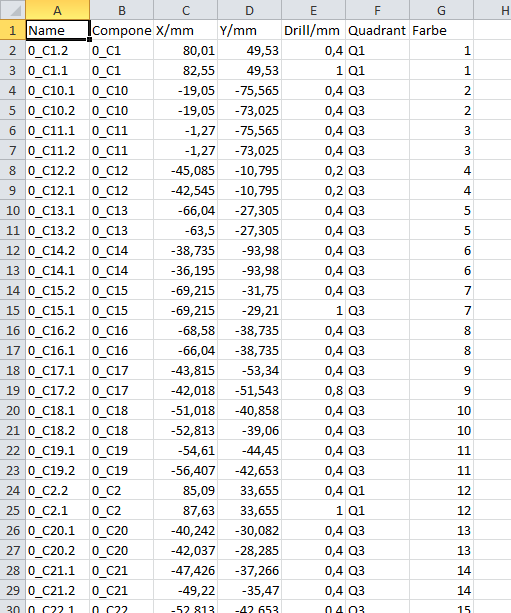
* Bild wird ausgerichtet durch ein Referenzbild
* Differenz vom Referenzbild und dem fehlerhaften Bild
* Durch eine Vorlage wird daraufhin bestimmt, um welches Bauteil es sich handelt

## Vorlage erstellen

Diese Vorlage muss für jeden Nadelkartentyp einmal erstellt werden und dazu werden die Koordinaten von den Bauteilen benötigt und eine so eine Vorlage.



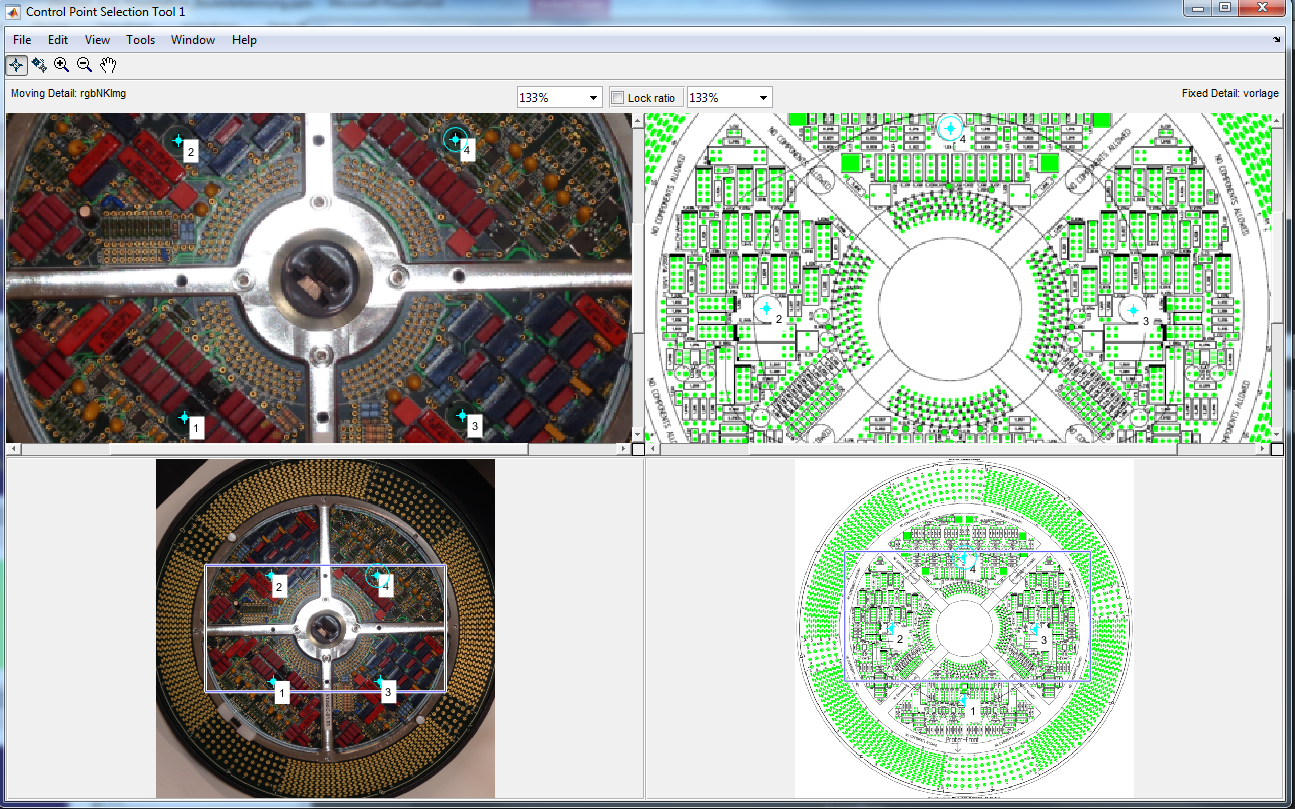
Beispiel der Koordinaten-Liste der Bauteile:



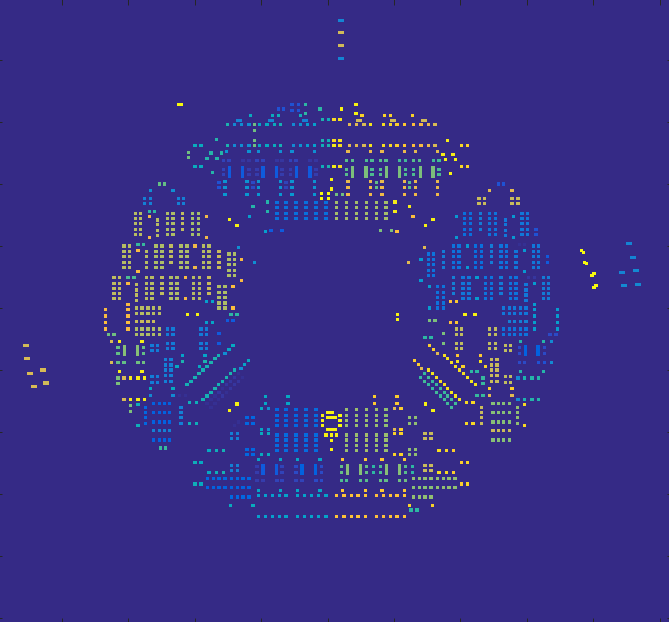
Außerdem wird das Referenzbild zugeschnitten und richtig ausgerichtet, so dass es auf die Vorlage passt. Beispiel für ein nicht zugeschnittenes Referenzbild:

**

Problem dabei ist, die Koordinaten sind ungenau und können alleine nicht verwendet werden und dazu dient die vorher gezeigt Vorlage. Zuallererst wird das zukünftige Referenzbild so ausgerichtet, dass es perfekt auf die Vorlage passt. Dazu werden diese Punkte verwendet, die vom User bestimmt werden müssen.

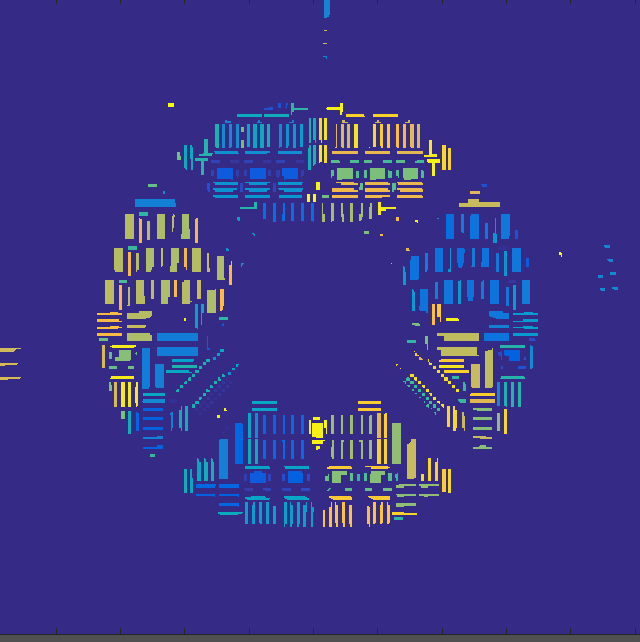


Daraufhin wird mit den Koordinaten ein „Komponenten Bild“ erstellt. Dieses sieht wie folgt aus.



Jedes Bauteil hat seine eigene Farbe und diese Farbe dient später zur Identifikation der Bauteile auf der Nadelkarte. Zu dieser Zeit sind es aber nur Punkte und dadurch werden die Punkte verbunden und leicht aufgeblasen:

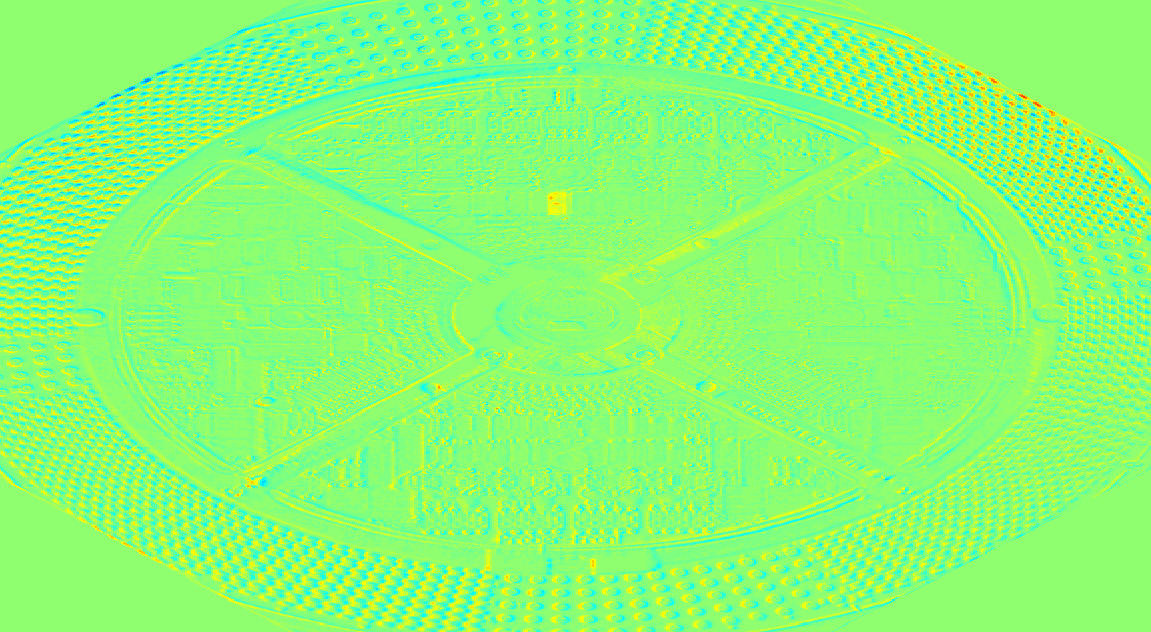


Danach wird diese Vorlage mit der ersten Vorlage multipliziert. Das dient dazu, dass nur mehr der Teil von den Rechtecken vorhanden ist der tatsächlich auf der Nadelkarte sein müsste:

Zum Schluss wird diese Vorlage und das ausgerichtete Referenzbild in den Data Ordner gespeichert. Dieser Data Ordner wird später für die Bauteil-Erkennung benutzt.

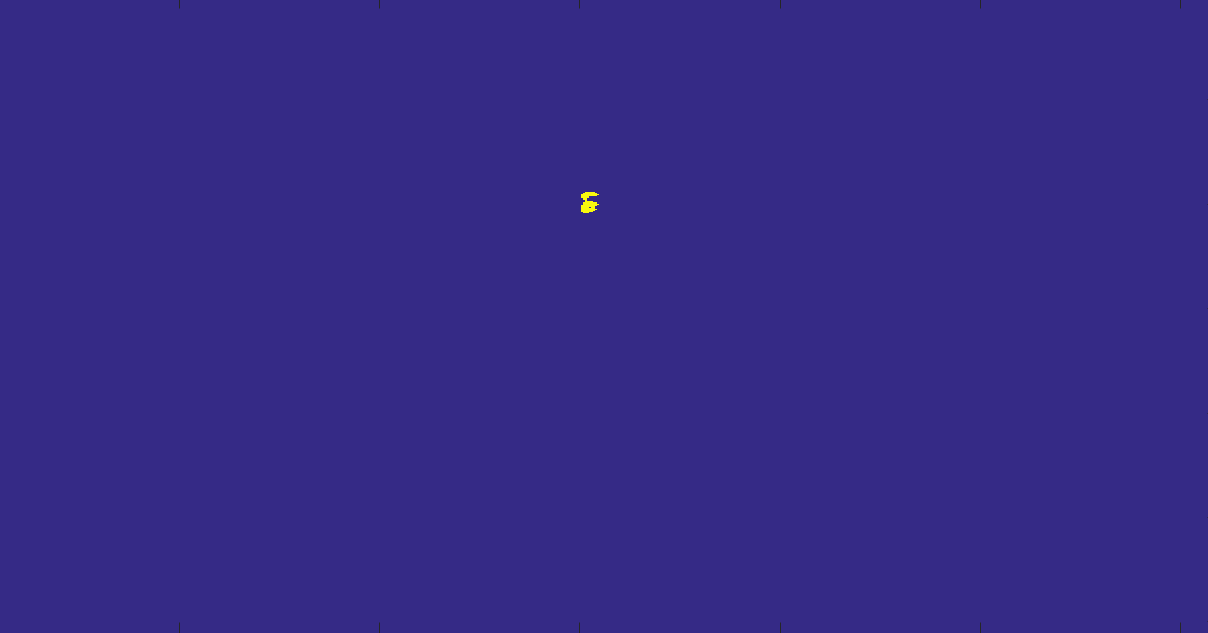
## Bauteile erkennen

Die beiden Bilder, also Referenzbild (Data Ordner) und das fehlerhafte Bild (User-Input) werden jetzt übereinandergelegt und daraufhin wird eine Differenz gebildet die wie folgt ausschaut:

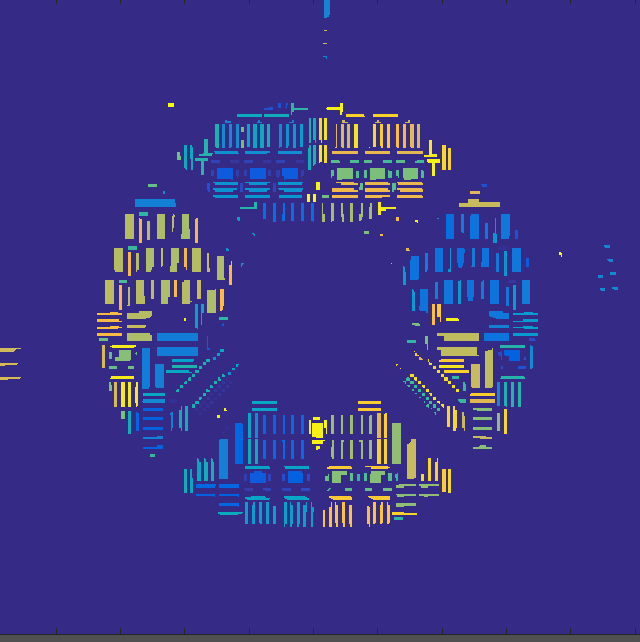


Hier kann man einen gelb/roten Fleck erkennen, welches das fehlende Bauteil ist. Es sind zwar außen noch andere Flecken, diese sind aber nicht wichtig und fallen im nächsten Schritt weg. Dieser Schritt ist die Herausnahme von den Pixeln die über einem Wert liegen. Dieser Wert ist die Genauigkeit die vom Benutzer eingegeben wird.

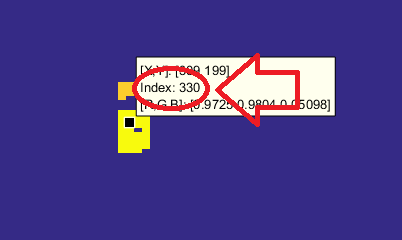
Daraufhin sieht das Bild so aus:



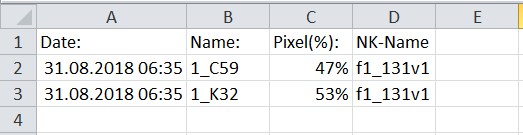
Hier sieht man nur mehr das Bauteil. Um es zu identifizieren wird dieses Bild mit unserer vorher erstellten Vorlage (siehe Bild) multipliziert:



Daraufhin bleibt von der Vorlage nur noch der kleine Fleck übrig der einen gewissen Index hat und mit diesem Index kann man in der Excel Datei das Bauteil ausfindig machen.

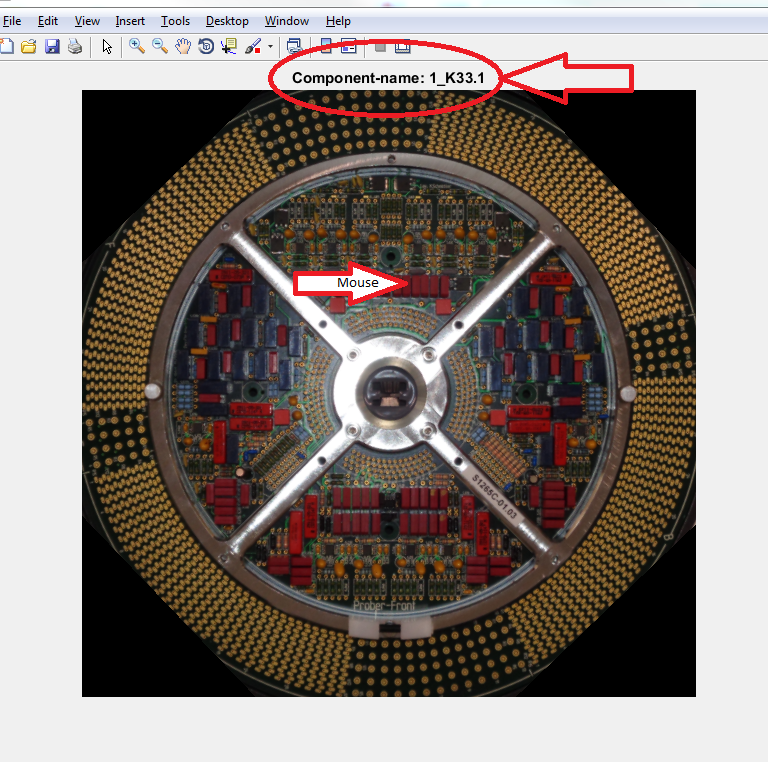


Diese Bauteile werden daraufhin in die Excel-Tabelle gespeichert:



## Vorlage Bauteile mousover

Ziel dieser Implementierung ist die Erkennung der Bauteile beim mouseover.



## Schluss

Die Bauteil-Erkennung ist zum jetzigen Zeitpunkt noch nicht zu 100% richtig, da die Vorlage, also genauer gesagt die Koordinaten ungenau sind und somit ist die Vorlage auch nicht präzise. Eine Verbesserung würde sich mit höherer Auflösung der Bilder bemerkbar machen. Außerdem ist die Vorlage-Erstellung nicht voll automatisch, da der User die Punkte bestimmen muss, um die das Referenz-Bild gedreht werden muss.