

# Echtzeit-Farbbildverarbeitung zur vollflächigen Kontrolle von Faltschachteln

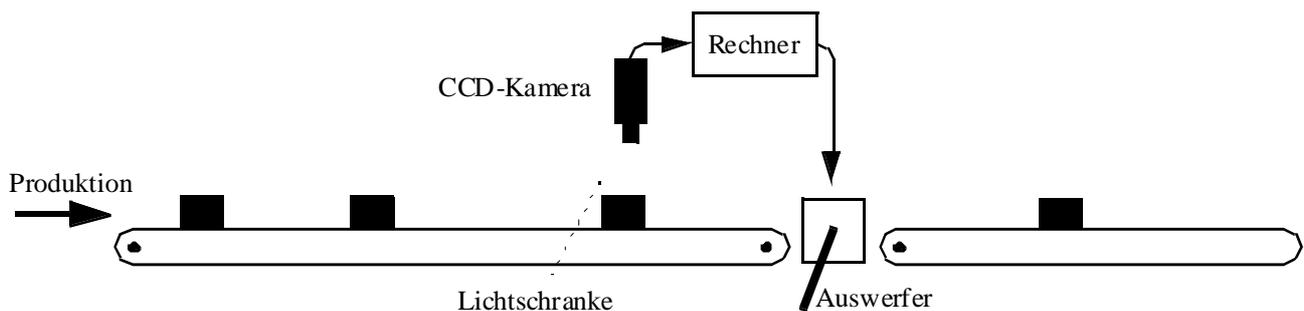
Jochen Biedermann und Ludwig Cromme  
Brandenburgische Technische Universität Cottbus  
Lehrstuhl für Numerische und Angewandte Mathematik  
Postfach 10 13 44  
D-03013 Cottbus  
Tel.: +49 355 69-2777  
e-Mail: cromme@math.tu-cottbus.de  
URL: <http://www.math.tu-cottbus.de/nam>

## Visuelle Qualitätssicherung in der Industrie

Im Bereich der industriellen Bildverarbeitung spielt die visuelle Qualitätssicherung eine entscheidende Rolle. Immer mehr Unternehmen setzen heutzutage Bildverarbeitungssysteme zur Qualitätssicherung ein oder denken zumindest über den Einsatz solcher Systeme nach [1].

Als Gründe hierfür sind neben der veränderten Angebotsseite durch die immer bessere Hard- und Software auch Veränderungen des Marktes zu nennen. Dazu zählen neben einem höheren Konkurrenzdruck und einer verstärkt betriebenen Automatisierung der Produktion in vielen Bereichen auch ein verändertes Qualitätsbewußtsein auf Seiten der Abnehmer (ISO/DIN 9000). Die Bereitstellung der Prüftechnik ist deshalb auch für Anbieter von Verpackungstechnik ein entscheidender Wettbewerbsfaktor geworden. Bei der Oberflächeninspektion sind insbesondere auch Prozeßoptimierungs- und Marketing-Aspekte wichtige Gesichtspunkte. Fehlerursachen sollen durch Rückkopplung der Inspektionseinrichtung mit dem Herstellungsprozeß schnell behoben werden und die optische Güte der Verpackung prägt zudem das „Image“ eines Produktes [2].

Schema eines automatischen visuellen Qualitätssicherungssystems



## Kontrolle von Faltschachteln

Die über den Handel zum Verbraucher gelangenden Waren in den Bereichen der chemischen, pharmazeutischen und Lebensmittelindustrie haben großen Einfluß auf Gesundheit und das Wohlbefinden des Menschen. Die Faltschachtel als Träger wichtiger Informationen über Identität, Wirkung und Anwendung eines Produktes spielt dabei eine wichtige Rolle. Der Hersteller hat sicherzustellen, daß diese Informationen beim Verbraucher ankommen und daß insbesondere Vertauschungen der Verpackung oder fehlerhafte Angaben ausgeschlossen sind, denn für den Endabnehmer ist dies häufig nicht kontrollierbar (z.B. bei Arzneimitteln). Für die amerikanische Pharmaindustrie ist deshalb heute schon laut der geltenden Gesetzgebung Vorschrift, daß die wichtigsten aufgetragenen Beschriftungen maschinell überprüft werden.

## Ausschnitt einer Arzneimittelpackung

**Verschreibungspflichtig**  
**Packungsbeilage**  
**beachten!**  
**1 Kapsel enthält:**

### Standard-Kontrollverfahren

Zur Zeit werden Faltschachteln überwiegend über Testmuster (Strichcodes oder ähnliche Markierungen) auf den Klebelaschen (Seitenlaschen) mit Farb-Punktsensoren kontrolliert (siehe Farb-Abb. 1 für ein typisches Kontrollmuster):

- Identitätskontrolle
- Kontrolle auf Mehrfachbedruckung
- Kontrolle auf Oberflächenfehler
- Farbkontrolle, ggf. Prägefolienkontrolle
- Positionierungskontrolle der Druckfarben
- Stanzkontrolle

Das Druck- und Erscheinungsbild der restlichen Faltschachteloberfläche bleibt dabei allerdings unberücksichtigt, d.h. es ist möglich, daß die Seitenlasche zwar korrekt bedruckt wurde, andere Flächen der Faltschachtel aber fehlerhaft ist.

Einige typische Fehler auf den Faltschachteln zeigt Farb-Abb. 2.

### Anforderungen an ein vollflächiges Kontrollsystem

Kontrolliert man dagegen die ganze Fläche, so sind die größten Probleme bei der Entwicklung das **industrielle Umfeld**, die **Geschwindigkeit** und die geforderte **Auflösung**.

In einer der Bedruckung und Stanzung nachgeschalteten Kontrolleinrichtung kann nicht von idealen Bedingungen ausgegangen werden, sondern Störungen durch kleinste Papierpartikel und Vibrationen müssen einkalkuliert und Positionierungsungenauigkeiten auf dem Fließband ausgeglichen werden. Faltschachteldruck, -stanz und -klebmaschinen können mit Geschwindigkeiten im Bereich von 5 bis zu 100 Schachteln pro Sekunde im Extremfall arbeiten. Ein deutlich langsames Kontrollsystem würde zu Problemen führen, da der gesamte Ablauf gebremst werden müßte. Die zu fordernde Genauigkeit der Kontrolle muß sich an der Wahrnehmbarkeit von kleinsten Fehlern bei der Bedruckung orientieren und liegt im Bereich von rund 100 bis 200 µm, was einer Auflösung von etwa 150 bis 250 dpi entspricht.

### Realisierung des vollflächigen Kontrollsystems

Durch den Einsatz von DSP<sup>1</sup>-basierter paralleler Rechentechnik im Zusammenspiel mit hocheffizienten Algorithmen zur Bildvorverarbeitung und -analyse wurde der Prototyp eines solchen Kontrollsystems für Faltschachteln entwickelt.

Das System ist skalierbar hinsichtlich der Rechenleistung durch den Einsatz von zusätzlichen Prozessorboards und ist nach oben hin nur durch die maximale Bildfrequenz des eingesetzten Kamerasystems von 30 Hz begrenzt.

Verwendet werden zwei asynchron triggerbare Progressive-Scan-CCD-Kameras: eine hochauflösende Schwarz-/Weiß-Kamera zur Verifikation der Detailtreue und eine Standard-Farb-CCD-Kamera, die Informationen über Farbton und -sättigung der

---

<sup>1</sup> Digitaler Signalprozessor

großflächigen homogenen Bereiche liefert. Für diese Zwei-Kamera-Lösung konnten in einer Machbarkeitsstudie deutliche Kostenvorteile (gegenüber einer einzelnen hochauflösenden Farbkamera) sowie eine erhebliche Reduktion der anfallenden Datenmenge ermittelt werden.

Nach der Bildaufnahme und einer optionalen Farbraumtransformation in den bezüglich der menschlichen Farbwahrnehmungen näherungsweise gleichabständigen CIE-Lab-Farbraum (alternativ kann aus Geschwindigkeitsgründen nichtlineares RGB verwendet werden) werden Bildhintergrund und Objekt automatisch erkannt und eine subpixelgenaue Lagekorrektur durchgeführt. Fehlerschranken werden durch Training mit Gut- und Schlechtbildern gelernt. Bei der Kontrolle werden zusätzlich zum Gesamtbild und Detailausschnitten auch die homogenen Farbbereiche sowie auftretende Falz- und Stanzkanten intensiv geprüft.

Die Farbanalyse und –kontrolle orientiert sich dabei hinsichtlich Auflösung und Toleranzen an den Eigenschaften des menschlichen Sehsystems, so daß keine Rechenzeit damit verschwendet wird, für den Menschen nicht wahrnehmbare Bildunterschiede festzustellen. Farb-Abb. 3 zeigt die Benutzeroberfläche der zugehörigen Prüfsoftware, die auf einem Windows-NT-Rechner läuft und unter Visual C++ entwickelt wurde. Durch Multithreading-Techniken konnten wesentliche Teile des gesamten Kontrollvorgangs über die natürliche Parallelität des eingesetzten DSPs hinaus parallelisiert werden.

### Farbraumtransformation nach CIE-Lab

Der dreidimensionale CIE-Lab-Farbvektor läßt sich aus einem Vektor  $(X, Y, Z)$  des Normfarbsystems bei vorgegebenem idealem Weiß  $(X_w, Y_w, Z_w)$  als Referenzweiß folgendermaßen ableiten [6]:

$$L = 116 \sqrt[3]{\frac{Y}{Y_w}} - 16$$

$$a = 500 \left( \sqrt[3]{\frac{X}{X_w}} - \sqrt[3]{\frac{Y}{Y_w}} \right),$$

$$b = 200 \left( \sqrt[3]{\frac{Y}{Y_w}} - \sqrt[3]{\frac{Z}{Z_w}} \right)$$

wobei man die Vektoren des Normfarbsystems beispielsweise aus Farbvektoren des RGB-Systems durch eine einfache lineare Transformation erhalten kann [7].

Der Lightness-Kanal  $L$  modelliert dabei das menschliche Helligkeitsempfinden, der  $a$ -Kanal beschreibt den Grün-Rot-Anteil und der  $b$ -Kanal den Blau-Gelb-Anteil der Farbe.

Recht effizient läßt sich die Farbraumtransformation von RGB nach CIE-Lab durchführen, indem man obige Rechenvorschrift als

$$L = C_1 \sqrt[3]{Y} - 16$$

$$a = C_2 \sqrt[3]{X} - C_3 \sqrt[3]{Y}$$

$$b = C_4 \sqrt[3]{Y} - C_5 \sqrt[3]{Z}$$

schreibt, wobei die Konstanten  $C_1, \dots, C_5$  nur einmal berechnet werden müssen und man für die Berechnung der dritten Wurzel geeigneterweise eine Lookup-Table einsetzt. Insgesamt werden dann pro Bildpunkt inklusive der linearen Transformation von RGB ins Normfarbsystem nur 8 Additionen/Subtraktionen, 13 Multiplikationen und 3 Lookup-Table-Zugriffe benötigt.

## Verarbeitungshierarchie und eingesetzte Verfahren

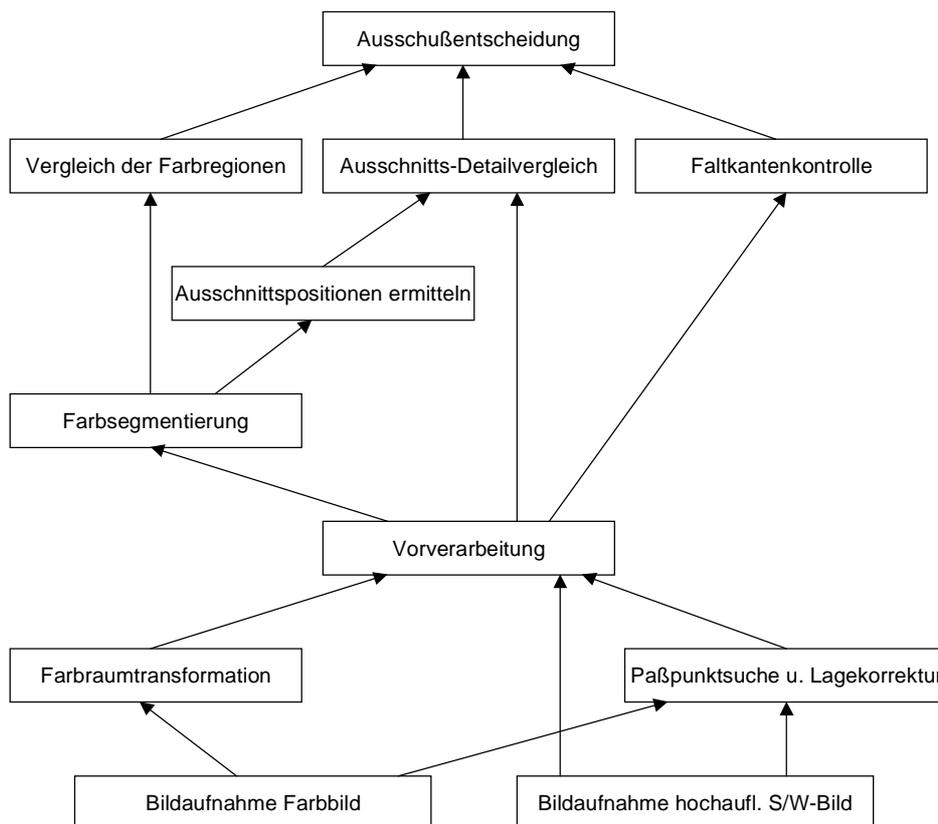
Bei der Planung und Entwicklung des Kontrollsystems wurde ein hierarchischer Ansatz gewählt, der die verschiedenen zu kontrollierenden Aspekte berücksichtigt und in der obersten Ebene mit einer Ausschußentscheidung endet. Es zeigte sich, daß insbesondere die Bereiche Lagekorrektur und Farbsegmentierung großen Einfluß auf Geschwindigkeit und Güte des Kontrollverfahrens haben.

Zur Lagekorrektur wird eine auf verschiedenen Auflösungsebenen operierende Top-Down-Suchstrategie mit anschließender Subpixel-Optimierung eingesetzt, die recht gute Ergebnisse liefert, aber in ihrer Laufzeit noch Verbesserungsspielraum besitzt. Erkannte Subpixel-Verschiebungen werden dann durch bilineare Interpolation ausgeglichen.

Bei der Farbsegmentierung fiel nach einer Analyse existierender Verfahren und möglicher Ansätze die Wahl auf zwei Strategien, die alternativ (je nach den vorliegenden Geschwindigkeitsanforderungen) eingesetzt werden können [3, 4]:

- Farb-Clusterverfahren oder
- Rekursives Histogramm-Splitting-Verfahren

Verarbeitungshierarchie des vollflächigen Kontrollsystems



## Farbsegmentierung durch ein Farb-Clusterverfahren

Beim Farb-Clusterverfahren werden im  $c$ -dimensionalen Farbraum  $\varepsilon$ -Umgebungen bzw. ihre Zentren  $w_i \in \mathbb{R}^c$ ,  $i = 1, \dots, n$ , „gelernt“ (z.B. durch Vorgabe von Gut- und Schlechtbilder).

Die eigentliche (zeitkritische) Segmentierung erfolgt dann über die vom Clusterverfahren gelieferte Abbildung  $\varphi : \mathbb{R}^c \rightarrow \{0, 1, \dots, n\}$ , die jedem Punkt des Farbraums die Nummer des Farbclusters  $(1, \dots, n)$  zuordnet bzw. 0, wenn der Punkt zu keinem Farbcluster gehört (Hintergrund etc.).  $\varphi$  wird dann auf jeden Bildpunkt angewandt. Dies kann sehr effizient durch den Einsatz einer geeigneten  $(c+1)$ -dimensionale Lookup-Table erfolgen, welche für große  $c$  und  $k$  allerdings sehr viel Speicherplatz in Anspruch nehmen kann. Man erhält so die  $n+1$  Regionen  $R_i = \{x \in E \mid \varphi(f(x)) = i\}$ ,  $i = 0, 1, \dots, n$ .

Der Rechenaufwand besteht hier im wesentlichen nur aus der Indexberechnung für die Lookup-Table in jedem Bildpunkt, was oft sogar von der Bildverarbeitungs-Hardware unterstützt wird.

## Farbsegmentierung durch rekursives Splitten von Farbkanalhistogrammen

Beim in der Grundidee auf Ohlander [5] zurückgehenden rekursiven Histogrammsplitting werden die Vorteile von rekursiven Splitting-Verfahren wie Split-and-Merge und von histogrammbasierten Verfahren für  $c$ -kanalige Farbbilder miteinander kombiniert. Die Grundidee besteht darin, alle  $c$  Farbkanalhistogramme des Bildes danach zu untersuchen, ob sie näherungsweise bimodal sind. Falls dies für mindestens eines der Histogramme zutrifft, wird für das hinsichtlich der Bimodalität beste Farbkanalhistogramm ein Schwellenwert bestimmt und das Bild so in zwei Teilbilder aufgeteilt. Diese Vorgehensweise wird dann rekursiv auf die entstandenen Teilbilder angewandt, solange bis kein so erhaltenes Teilbild ein näherungsweise bimodales Histogramm mehr besitzt.

Durch die Suche nach einem geeigneten Schwellenwert jeweils in den Farbkanalhistogrammen reduziert man das Problem von einem  $c$ -dimensionalen auf ein eindimensionales Schwellenwertproblem und kann die bekannten Verfahren zur Schwellenwertbestimmung einsetzen.

Die Bimodalität eines Histogrammes wird als eine Fuzzy-Aussage aufgefaßt und so quantisierbar gemacht:  $\mu : (K \rightarrow \mathbb{N}_0) \rightarrow [0, 1]$  sei dazu eine geeignet gewählte Zugehörigkeitsfunktion eines Histogrammes  $h : K \rightarrow \mathbb{N}_0$  zu der Menge der bimodalen Histogramme, wobei  $K = \{0, \dots, k\}$  der Wertebereich des Histogrammes bzw. des zugehörigen Bildes sei. Dabei werde vorausgesetzt, daß Histogramme  $h$  einpunktiger Punktmenge nicht bimodal sind, d.h.  $\mu(h) = 0$  für  $\sum_{i \in K} h(i) = 1$ .

Insgesamt bekommt man nun folgenden Grundalgorithmus:

**Algorithmus:** Gegeben sei ein  $c$ -kanaliges Farbbild  $f = (f_1, \dots, f_c) : E \rightarrow K$  auf dem endlichen Gitterrechteck  $E \subset \mathbb{Z}^2$  sowie ein  $\delta \in [0, 1]$ .

1. Setze  $n = 0$  und erzeuge eine Liste  $L$  von Punktmenge:  $L = \{E\}$

2. Wähle eine beliebige Punktmenge  $B$  aus der Liste  $L$ .

Bestimme die Farbkanalhistogramme  $h_i : K \rightarrow \mathbb{N}_0$ ,  $i = 1, \dots, c$  des Farbbildes  $f|_B$ .

Bestimme (ein)  $j$  mit  $\mu(h_j) = \max_{i=1, \dots, c} \mu(h_i)$ .

Falls  $\mu(h_j) \geq \delta$  :

Bestimme einen Schwellenwert  $t \in K$ , splitte  $B$  in die zwei disjunkten und

nichtleeren Punktmengen  $B_1 = \{x \in B \mid f_i(x) \geq t\}$  und  $B_2 = \{x \in B \mid f_i(x) < t\}$  und füge diese zur Liste  $L$  hinzu.

Ansonsten setze  $R_{n+1} = B$  und erhöhe  $n$  um 1.

3. Entferne  $B$  aus der Liste  $L$ .
4. Falls die Liste  $L$  noch Punktmengen enthält, gehe zu 2.
5.  $R_1, \dots, R_n$  sind die segmentierten Regionen.

Es läßt sich zeigen, daß dieser Grundalgorithmus endlich ist und sich der Rechenaufwand des Verfahrens nach oben durch

$$(3n - 2)|E| + c(2n - 1)(a(k) + 1) + (n - 1)b(k)$$

Rechenoperationen abschätzen läßt, wobei  $a(k)$  die Anzahl der zur Bestimmung von  $\mu(h_i)$  und  $b(k)$  die Anzahl der zur Schwellenwertbestimmung notwendigen Rechenoperationen seien.

### Ausblick

Mit dem ersten Prototypen konnte ein Durchsatz von 8 Faltschachteln pro Sekunde erreicht werden und die meisten der bekannten Fehlertypen wurden erkannt.

Bei den durchgeführten Test zeigte sich, daß Windows-NT als Betriebssystem (wie auch nicht anders erwartet) für Echtzeitanwendungen nicht sonderlich geeignet ist, da bei sehr hohen Geschwindigkeiten immer wieder kurze Unterbrechungen auftraten. Als Alternative ist der Einsatz eines Echtzeit-Betriebssystems (z.B. RTOS) geplant.

Weiterhin wird eine Erhöhung des Durchsatzes durch Code-Optimierung und die DSP-nahe Programmierung von zeitkritischen Passagen angestrebt sowie das System im industriellen Einsatz getestet.

Ein drittes Ziel ist die Verringerung der Systemkosten durch die Verwendung alternativer Kameras, um so die Marktchancen des Systems weiter zu erhöhen.

### Literatur

- [1] „Produkte und Dienstleistungen für die Bildverarbeitung – Stand und Trends“, Studie des Ministeriums für Wirtschaft und Mittelstand, Technologie und Verkehr des Landes Nordrhein-Westfalen, Düsseldorf, 1996
- [2] „Industrielle Bildverarbeitung / Machine Vision“, Fachgemeinschaft Robotik und Automation des VDMA, 3. Auflage, 1997
- [3] J. Biedermann. „Analysis of Segmentation Strategies in Real Time Color Image Processing for Industry“, Proceedings of the GAMM Annual Meeting, Bremen, 1998
- [4] J. Biedermann. „Numerische Algorithmen zur Echtzeit-Farbbildverarbeitung mit Anwendungen in der industriellen Qualitätssicherung“, Dissertation, Brandenburgische Technische Universität Cottbus, 1999
- [5] R. Ohlander, K. Price und D.R. Reddy. „Picture segmentation using a recursive region splitting method“, Computer Graphics and Image Processing 8, S. 313-333, 1978
- [6] M. Richter. „Einführung in die Farbmétrie“, Walter de Gruyter Verlag, Berlin, New York, 1981
- [7] H. Ernst. „Einführung in die digitale Bildverarbeitung“, Franzis-Verlag, München, 1991

Abb. 1: Kontrollfeld auf der Seitenlasche einer Faltschachtel

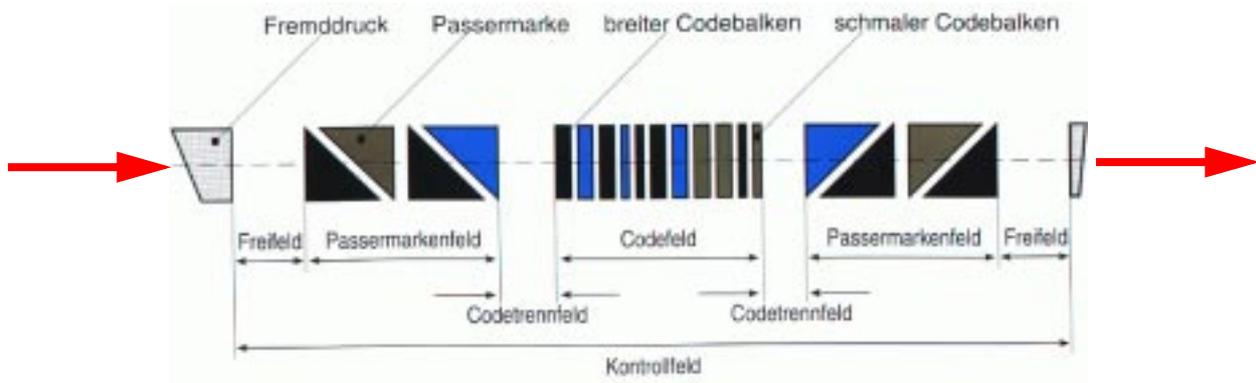
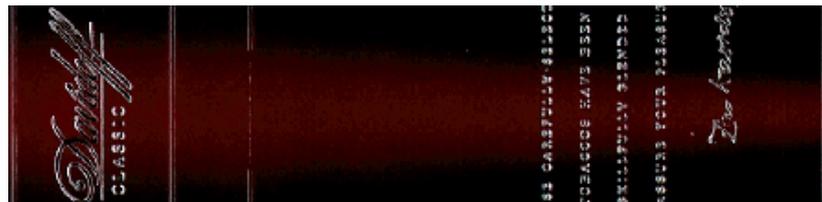


Abb. 2: Einige Fehlertypen

- Materialfehler



- Falscher Farbton

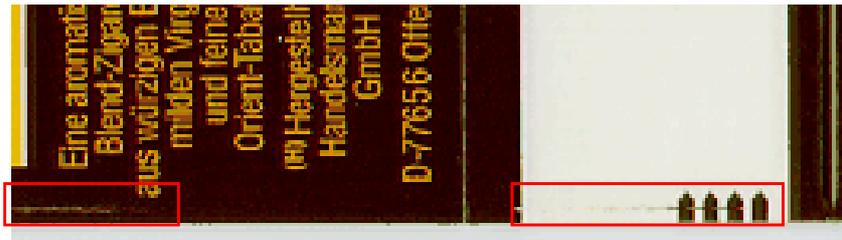


- Falsche Bedruckung



Abb. 2 (Fortsetzung):

- Verschmierungen



- Farbpasser



Abb. 3: Benutzeroberfläche des Kontrollprogramms

