

Vollständige Oberflächenprüfung von Orangen mittels Farbbildverarbeitung

Karl-Heinz Franke¹, Daniel Kapusi¹, Carsten Lucht², Rico Nestler², Dietmar Kollhoff³

¹ Technische Universität Ilmenau, Fachgebiet GDV
PF 100565, D-98694 Ilmenau

² Zentrum für Bild- und Signalverarbeitung e.V.

³ Gesellschaft für Bild- und Signalverarbeitung mbH
Gustav-Kirchhoff-Str. 5, D-98693 Ilmenau
info@zbs-ilmenau.de bzw. info@gbs-ilmenau.de

1 Einleitung

Die visuelle Qualitätsprüfung von Agrarprodukten in verschiedenen Verarbeitungsstufen stützt sich seit jeher besonders auf die Auswertung von Farbmerkmalen. Es ist deshalb folgerichtig, dass der Erfassung und der Verarbeitung von Farbinformation auch bei der Automatisierung der Qualitätsprüfung dieser Produkte eine große Bedeutung zukommt. Die Nutzung der Farbinformation bleibt dabei nicht auf die Klassifikation beschränkt, sondern aus der Bewertung der Ergebnisse werden gleichzeitig Hinweise für die Produzenten oder für die Verarbeitung angeleitet. In diesem Beitrag soll die mehrseitige Nutzung der Farbinformation am Beispiel der Oberflächeninspektion von Orangen diskutiert und veranschaulicht werden.

2 Problemstellung bei der Kontrolle von Zitrusfrüchten

Aus dem äußeren Erscheinungsbild der Orangen oder ähnlicher Zitrusfrüchte kann zu einem hohen Prozentsatz auf ihre Qualität geschlossen werden. Mangelerscheinung äußern sich in Verfärbungen (z.B. wässrige Fäule, schwarzer oder weißer Schimmelbefall), Deformationen (z.B. Grapefruitform bei Orangen oder mechanische Schäden) und Veränderungen der Oberflächentextur (z.B. Großporigkeit). Außerdem kann aus der Farbe auf den Reifegrad und erforderliches Degreening geschlossen werden oder sie kann der Schätzung des Zuckergehaltes im Saft dienen.

Die Fehler sind oft nur in lokal begrenzten Arealen zu beobachten, so dass für eine leistungsfähige Inspektion die vollständige Erfassung der Fruchtoberfläche erfolgen muss. Die Vermeidung von Okklusionen erfordert eine nahezu gleichzeitige Aufnahme mehrerer Ansichten im freien Fall. Für die Betrachtung konvexer Objekte im d -dimensionalen Objektraum ergeben sich für ebene Projektionen zwingend Anordnungen im $(d+1)$ -Simplex, für $\mathbb{R}^3$ also im 4-Simplex, wenn die Anzahl der erforderlichen Kameras minimiert werden soll. Auf die konkrete Beleuchtungs- und Aufnahmeanordnung sowie ihre Integration in den logistischen Ablauf der Früchteverwertung soll aus inhaltlichen Gründen nicht näher eingegangen werden. Der Problemkreis wurde aber erwähnt, weil daraus besondere Anforderungen an die Verarbeitungsgeschwindigkeit und, in Verbindung damit, an die Soft- und Hardwarearchitektur gestellt werden.

Anhand der Bildaufnahmen wird die Orange mit hocheffizienten Algorithmen und leistungsfähiger Rechen-technik klassifiziert, so dass eine Ausschleusung von mangelhaften Früchten im Auffangbereich unmittelbar nach der Fallstrecke erfolgen kann. Zur Inspektion wird ein mehrstufiger Klassifikationsprozess genutzt, wofür Farb-, Textur- und Formmerkmale berechnet werden. Die Farbinformation wird dabei mit verschiedenen Zielstellungen eingesetzt. Zur Unterscheidung verschiedener Klassen von Oberflächenerscheinungen wird in der ersten ikonischen Klassifikationsstufe ein zwei- bis dreidimensionaler Farbvektor genutzt.

Die in einigen Applikationen eingesetzte Texturklassifikation bedient sich der Farbbildverarbeitung, indem Farbprojektionen zur gezielten Verstärkung und/oder Unterdrückung von Texturen genutzt werden. Ziel ist dabei eine Trennung der Farbtexturen (z.B. Farbmaserungen) von räumlichen Strukturen.

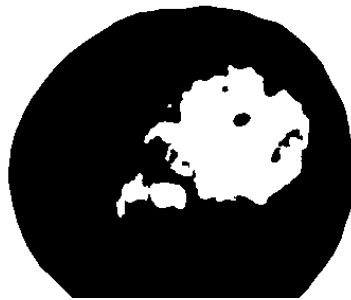
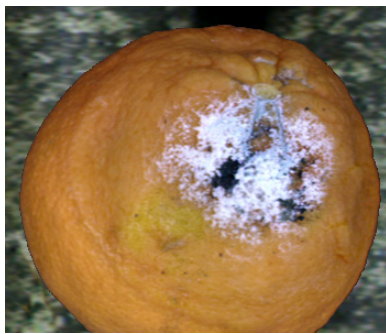
Die Konstanz des grundsätzlichen Bunttonwinkels wird zur einfachen Freistellung der Orange (Objektsegmentierung) genutzt. Aufwendige Sonderfälle können durch eine zusätzliche farbliche Hintergrundgestaltung der Fallstrecke vermieden werden.

Schließlich dient die Farbe der befallenen Oberflächenbereiche zum Aufbau von Statistiken für die Erzeuger (Auswahl von Maßnahmen zur Bekämpfung von Pflanzenkrankheiten) oder zur Verarbeitung (z.B. Degree-ning).

3 Klassifikation befallener Oberflächenareale

3.1 Freistellung der Orangen

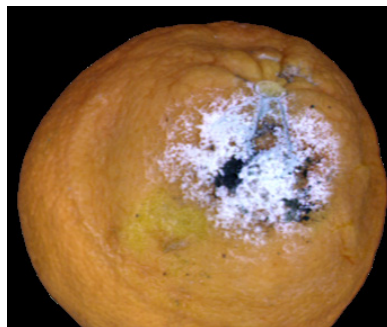
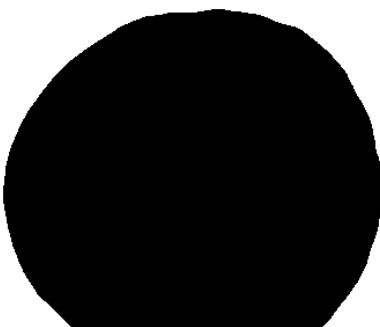
Je nach Auslegung der Sortierstation erfolgt die Einzelbildaufnahme vor dem Hintergrund von Transportrollen, speziellen Transportbehältnissen oder der Fallstrecke. Dieser Hintergrund ist weitgehend undefiniert, so dass die einzelnen primären Farbkanäle bzw. Intensität oder Helligkeit keinen einfachen Ansatz zur Freistellung der Objekte liefern. Bei Orangen fällt jedoch ins Auge, dass der Bunttonwinkel (Hue) der gesunden Schale sehr stabil ist und i. Allg. einen hinreichenden Abstand zu den praktisch auftretenden Hintergründen aufweist. Auszuschließen sind jedoch dunkle Hintergrundbereiche, denn bekanntlich ist der Bunttonwinkel für kleine Intensitäten sehr instabil bzw. unbestimmt.



Die auf diese Weise extrahierbaren Objektmasken sind trotz morphologischer Vor- und Nachbehandlung oft noch mit Löchern versehen, die durch schadhafte Bereiche der Schale verursacht werden. Die Farbvalenzen der potentiellen Schadstellen (schwarzer und weisser Schimmel, Fäule, Schorf, mechanische Schäden) können nämlich durchaus mit

den Hintergrundvalenzen zusammenfallen.

Da die Stärke und Art des Befalls festzustellen sind, müssen die Fehlstellen der Masken geschlossen oder ihr Entstehen von vornherein vermieden werden. Letzteres kann bei Beobachtung im freien Fall durch Schaffung eines Hintergrundes erreicht werden, der in Hue hinreichenden Abstand zur Schale und zu allen

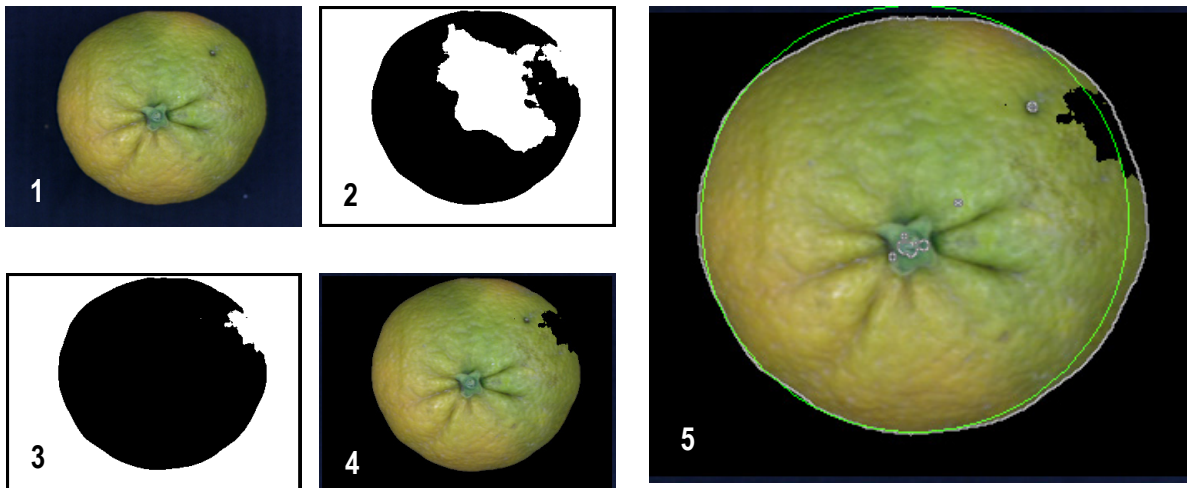


Fehlstellenvalezen aufweist (Blau, Cyan bis zu gesättigtem Grün sind möglich). Durch zusätzliche Auswertung der Sättigungskomponente lassen sich dann auch nahezu unbunte Oberflächenareale (Schimmel, Schorf) zuverlässig vom Hintergrund separieren.

Für „natürlichen“ Hintergrund des Transportbandes bzw. der Werkhalle sind die Löcher zu detektieren (Finden innerer,

zusammenhängender Komponenten z.B. mit einem Zeilenkoinzidenzverfahren) und zu schließen. Problematisch sind dabei „Randberührungen“ der Maskenfehlstellen, die zu erheblichen Konkavitäten des Maskenrandes führen können.

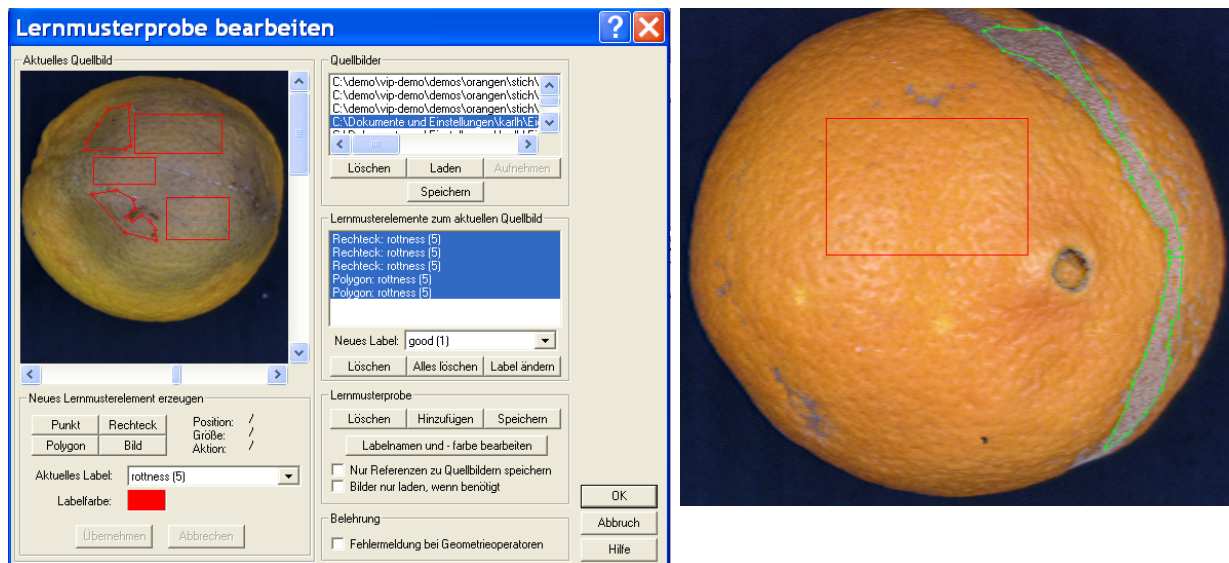
Die Houghtransformation für Kreise oder morphologisch aufbereitete Konturen helfen in diesen Fällen, die Objektmaske zu schätzen.



3.2 Harte ikonische Klassifikation, partitionierende Clusterung und die Verwendung von Membershipfunktionen auf der Pixelebene

3.2.1 Klassen für die Oberfläche

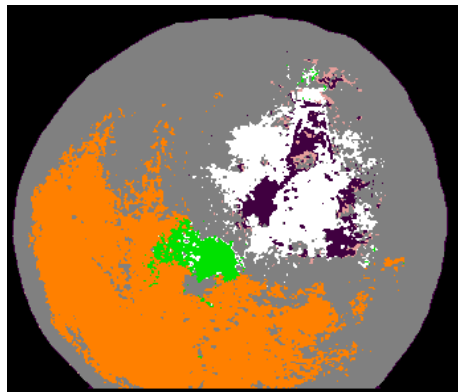
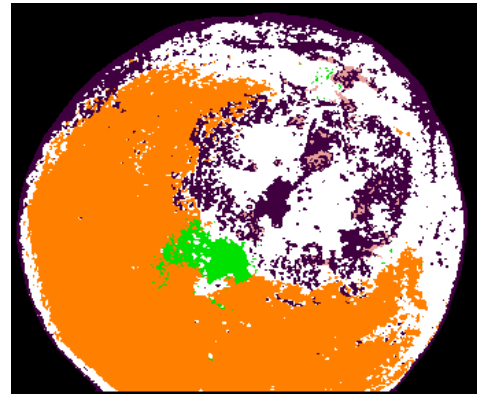
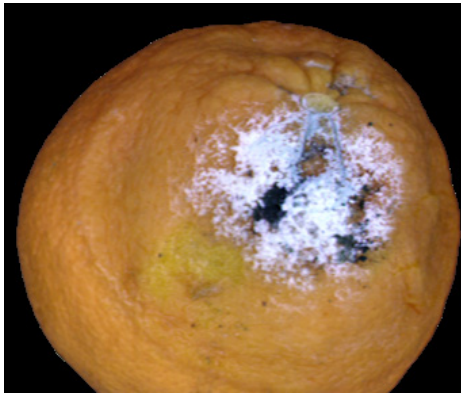
Beispiele für Oberflächenregionen von Orangen, die als „gut“ (K1), „weißer Schimmel“ (K2), „schwarzer Schimmel“ (K3) sowie „grün“ (K4) befunden werden, sind in Abschnitt 3.1 dargestellten Bildern ersichtlich. Daneben sind noch die Klassen „Fäule“ (K5) und „Schorf“ (K6) von Bedeutung.



Auf die genannten gesunden und schadhafte Oberflächenregionen wurde mit Hilfe des dargestellten Belehrungsdialoges angelernt. Die Stichproben stammen von über 100 unterschiedlichen Orangen aus unterschiedlichen „Chargen“ (unterschiedliche Produzenten, unterschiedliche Anbauregionen, unterschiedliche Erntezeitpunkte, unterschiedliche Baumkulturen).

3.2.2 Harte Maximum-Likelihood-Klassifikation

Im Wesentlichen stehen für die Klassifikation geometrische, abstandsmessende und statistische Methoden sowie Rekonstruktionsverfahren zur Verfügung. Die Form der Klassenbeschreibungsräume ist für die diskutierte Anwendung unbekannt, die erzeugbaren Stichproben besetzen diese Räume stark heterogen und die Stichproben sind von stark unterschiedlichem Umfang. Aus diesen und weiteren Gründen (z.B. Speicherung, Nachbelehrung, Kombination von Lernergebnissen) wurde für die harte Klassifikation zunächst der Maximum-Likelihood-Klassifikator gewählt.



Labelfarben:

- gut
- weißer Schimmel
- schwarzer Schimmel
- grün
- Fäule
- Schorf

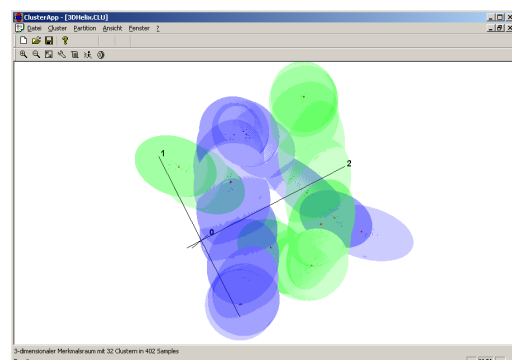
Bei Nutzung der Klassen K1 bis K5 halten sich die Überlappungen der Beschreibungsräume in Grenzen, so dass befriedigende Ergebnisse erzielt werden. Die Fehlklassifikationen am Rande können wegen der Beleuchtungsabschattung toleriert werden (durch Ausblenden aus dem Einzelbild). Sehr schlecht werden aber die Ergebnisse unter Hinzunahme der Klasse „Schorf“ (K6). Das liegt an der großen Streubreite dieser Klasse, die bei Approximation mit einem Ellipsoid zu extremen Überlappungen der approximierten Klassenbereiche im Farbraum führt. Eine Verbesserung der Klassenbeschreibung wird offensichtlich erforderlich.

3.3 Verwendung von partitionierenden Clusteralgorithmen zur Verbesserung der Klassenbeschreibungsräume

3.3.1 Ansatz und Zielstellung der Clustering

Obwohl der Bayessche Grundansatz neben entscheidungsspezifischen Kosten und den a-priori-Wahrscheinlichkeiten beliebige Dichten für die klassenbedingte Verteilung der Merkmalvektoren (hier Farbvalenzen) zulässt, werden aus praktischen Gründen oft multivariate Gaußdichten zur Approximation der Klassenbeschreibungsräume eingesetzt. So auch im vorangegangenen Abschnitt.

Ist die Variabilität in den Farbvalenzen jedoch nicht vordergründig durch gaußförmige Störgrößen sondern eher durch objektbedingte Variabilität bestimmt, wie sie für Naturpro-



dukte typisch ist, muss von komplexer Form der Klassenbeschreibungsräume ausgegangen werden. Diese komplexe Form ist i. Allg. nicht explizit bekannt, sie kann aber oft aus einer begrenzten Zahl vorklassifizierter Beispieldaten (Samples) mehr oder minder gut geschätzt werden.

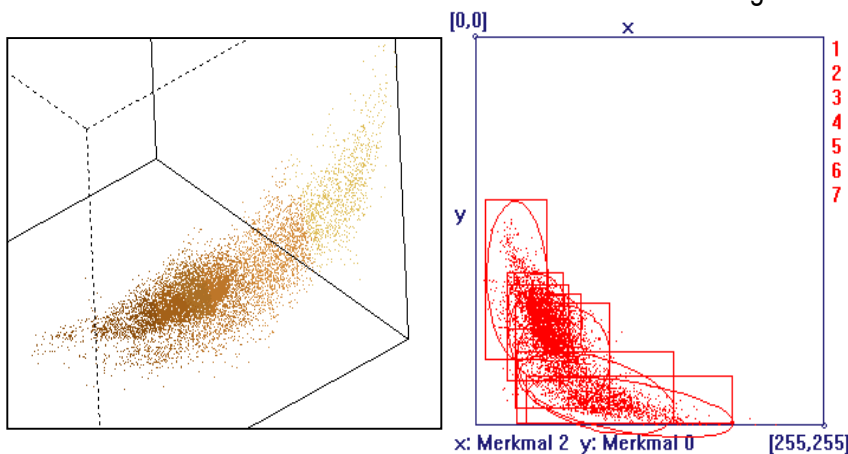
An diesem Punkt setzen Clusteralgorithmen an. Sie dienen der automatischen Nachbildung des Sampledatensatzes durch ein Set hyperelliptischer Primitive (Cluster). Diese Clustermenge, die Partition, stellt eine generalisierte Beschreibung der durch die Samples belegten Klassenbereiche dar und ermöglicht so die effiziente parametrische Klassifikation von beliebigen Merkmalsvektoren. Die parametrische Beschreibung gewährleistet zudem die kompakte Speicherung der Klassenbereiche.

Im VIP-Toolkit bzw. der von ZBS e.V. angebotenen ClusterLIB-DLL¹ sind für m-dimensionale Merkmalräume zwei Verfahren zur Erstellung und Optimierung von Partitionen implementiert: die Radiusrestriktionsmethode und das k-Means-Austauschverfahren. Durch die Kombination dieser Verfahren werden hinsichtlich Einfachheit und Kompaktheit optimale Partitionierungsergebnisse garantiert. Beide Algorithmen sind robust und einfach zu parametrieren.

Mit Hilfe der für jede Klasse erstellten Partition können anschließend beliebige Merkmalsvektoren klassifiziert werden. Dies geschieht üblicherweise mit einem Minimaldistanzklassifikator anhand der parametrischen Beschreibung der Partition. Darüber hinaus besteht die Möglichkeit, zusätzlich verfügbares Wissen durch Vergabe von a-priori-Wahrscheinlichkeiten für jede Klasse (oder wahlweise auch für einzelne Cluster) einzubringen und damit einen Bayes-Klassifikator zu realisieren.

3.3.2 Anwendung bei der Pixelklassifikation von Schadstellen auf Orangenoberflächen

Der beschriebene Clusteralgorithmus wurde auf das Problem „Orangen“ angewendet, wobei für die vorgeschaltete Radiusrestriktionsmethode für jede Klasse 5 bis 7 Cluster pro Partition gesetzt und die Zahl der Samples auf 10000 beschränkt wurden, wenn dies die Stichproben hergeben. Für das sich anschließende k-Means-Austauschverfahren wurden maximal 6 Iterationen eingestellt. Die Clustertrennung erfolgte mittels Bayesscher Trennfunktion.



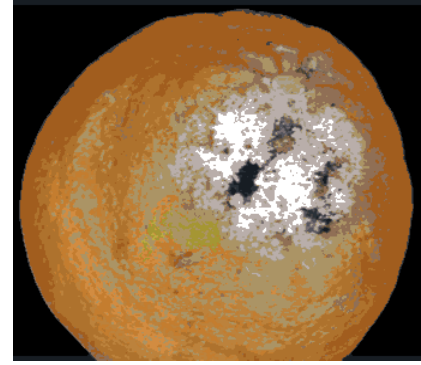
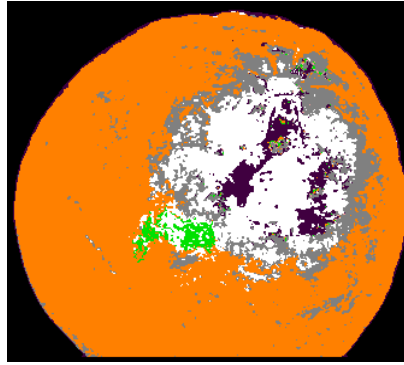
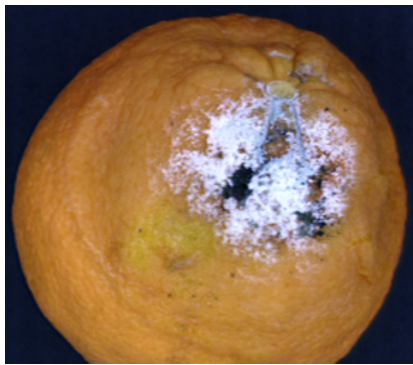
Bereits die Betrachtung der Punktwolken einzelner Klassen lässt erkennen, dass die Beschreibung der Einzelklasse mit einer multivariaten Gaußverteilung diese nur grob approximiert.

Für die farbliche Darstellung der Punktwolke wurden die Schwerpunktvalenzen der Teilcluster der Partition für die Gut-

klasse verwendet.

Wesentlich näher kommt das Set von Clustern (rechts: siehe eine Projektion des 3D-Farbraumes), das der Clusteralgorithmus liefert und das für das Gutcluster rechts dargestellt ist.

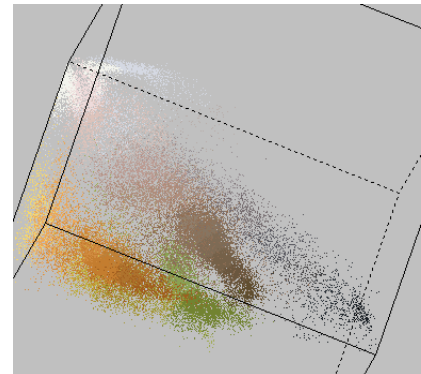
¹ Weitere Infos: www.zbs-ilmenau.de/html/prod/cluster.html,
Download: www.zbs-ilmenau.de/intern/cluster/cluster.html



Die gute Erfassung der Klassenbeschreibungsräume führt zu wesentlich besseren Klassifikationsergebnissen (Bild in der Mitte). Rechts oben sind die mittleren Farbvalenzen der Teilcluster für die Partitionen der sechs Klassen eingetragen und rechts unten ist ein Blick in den Farbraum mit den für die Clusterung verwendeten Samples gegeben.

Labelfarben:

- gut
- weißer Schimmel
- schwarzer Schimmel
- grün
- Fäule
- Schorf



3.4 Nutzung von Membershipfunktionen auf ikonischer Ebene

Eine harte Klassifizierung führt aufgrund der pixelweisen Zuordnung zu genau einem Klassenlabel (Quantisierungsrauschen) und ohne jede Berücksichtigung der lateralen Beziehungen zu den Nachbarpixeln oft zu einem Informationsverlust. Die Verwendung von Membershipfunktionen und die Einbeziehung lateraler Zusammenhänge bildhafter Information stellt einen Ansatz zur Überwindung der daraus resultierenden Probleme dar.

3.4.1 Erzeugung von Membershipbildern

Jedem Pixel wird auf ikonischer Ebene ein Vektor zugewiesen, dessen Elemente die Auftretenswahrscheinlichkeiten des Inputsamples für die einzelnen Klassen beinhalten. Diese Vorgehensweise lässt sich unter vereinfachenden Annahmen aus dem Bayesschen Grundansatz ableiten.

Es ist also Kenntnis darüber erforderlich, wie sich die Merkmalvektoren $\underline{X}$ bei Vorliegen der einzelnen Klassen C_k im Merkmalraum verteilen. Diese klassenbedingten Wahrscheinlichkeitsdichten $w(\underline{X}/C_k)$ sind im Sinne des Bayesschen Ansatzes zunächst beliebig, werden aber oft mittels multivariater Gaußdichten aus einer Stichprobe heraus modelliert (n – Dimension des Merkmalsraums, $\underline{KK}_k$ – Kovarianzmatrix von C_k , d_{mhl} – Mahalanobisabstand):

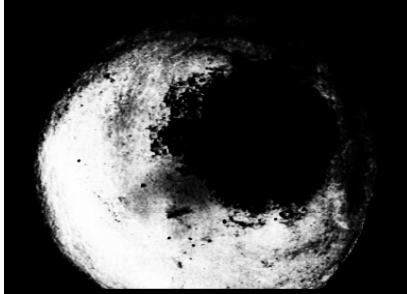
$$w(\underline{X}/C_k) = \frac{1}{\sqrt{(2\pi)^n} \cdot \sqrt{\text{Det}(\underline{KK}_k)}} \cdot e^{-\frac{1}{2} \cdot d_{mhl}(\underline{X}/C_k)} \quad 3-1$$

Komplexere Dichten lassen sich, wie im Kapitel 3.3 (Clusterung) erläutert, durch ein Set solcher Dichten approximieren.

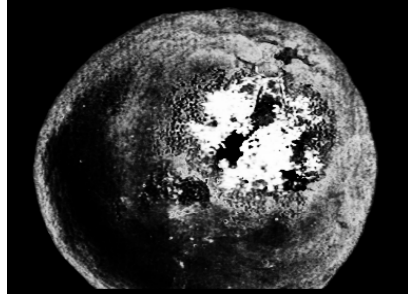
Im Sinne der Bayesschen a-posteriori-Wahrscheinlichkeiten wird für jede Klasse k ein Membership-Bild erzeugt, welches unter der Annahme, dass der Inputvektor an der entsprechenden Pixelposition Element dieser Klasse k ist, die Wahrscheinlichkeitsdichte für das Auftreten dieses Inputvektors repräsentiert.

$$P(C_k / \underline{X}) \propto w(\underline{X} / C_k) \quad \text{falls } P(C_i) = P(C_j) \forall i \neq j \quad 3-2$$

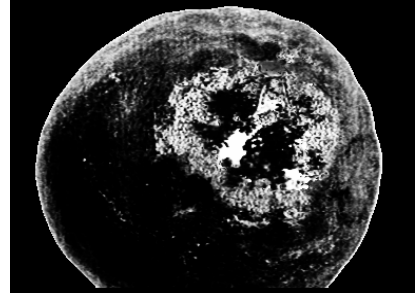
Dieses Membershipbild wird auf die unbedingte Wahrscheinlichkeitsdichte $w(\underline{X})$ normiert und für Darstellungs- und Verarbeitungszwecke auf den Wertebereich des verwendeten Bilddatentyps (im Beispiel 255) skaliert. Implizit wurde dabei angenommen, dass die bedingten Dichten für alle Klassen gleich zu wichten sind (d.h. gleiche bedeutungsgewichtete a-priori-Wahrscheinlichkeit der Klassen).



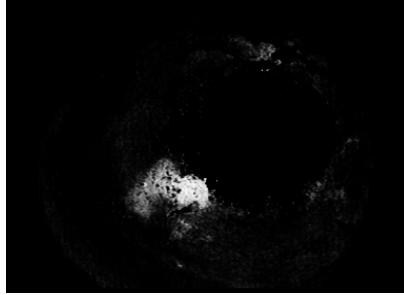
Membershipbild Label 1 (gut)



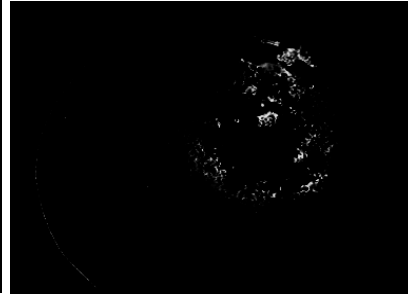
Membershipbild Label 2 (weißer Schimmel)



Membershipbild Label 3 (schwarzer Schimmel)



Membershipbild Label 4 (grün)



Membershipbild Label 5 (Fäule)

3.4.2 Bayessche Filterung

Auf Basis der Membershipbilder erfolgt die Filterung über den Bayesschen Ansatz unter Berücksichtigung der lateralen Pixelbeziehungen mit dem Ziel, Ausreißer zu eliminieren und eine Glättung im Labelbild zu erreichen.

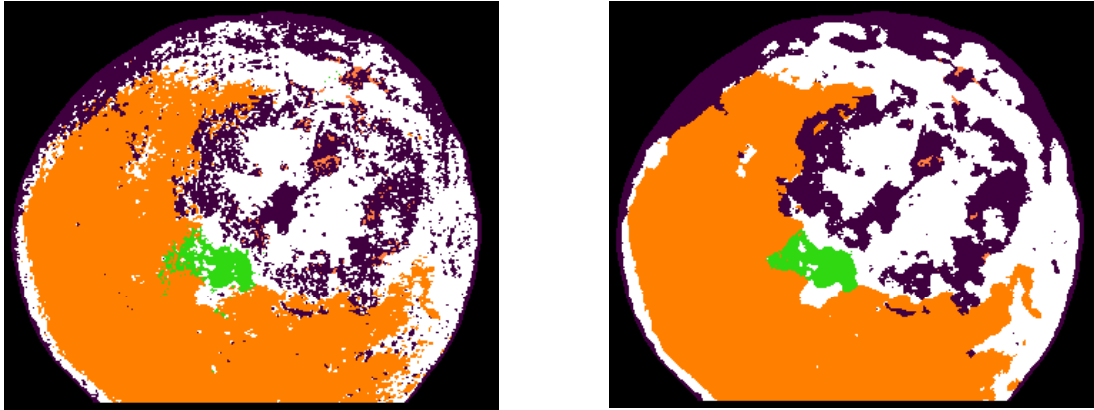
Aus den N Pixeln der Nachbarschaft U des Aufpunktes P_0 wird mittels Gleichung 3-3 die mittlere Auftrittswahrscheinlichkeit der Klassen bestimmt. Diese steht, ähnlich der a-priori-Wahrscheinlichkeiten der Klassen (Prior) beim Gesamtproblem, für die Bedeutung der jeweiligen Klassen in der Region U . Sie soll als *lokaler Prior* bezeichnet werden, da, im Gegensatz zur eigentlichen Bedeutung der Priorwahrscheinlichkeit als globale, problembezogene und vor allem stichprobenunabhängige Kenngröße, die Gültigkeit und der Einfluss in diesem Fall aufgrund der Stichprobenmenge N nur lokal auf die Nachbarschaft des betrachteten Aufpunktes P_0 begrenzt ist.

$$P(C_k) \big|_{U(P_0)} = \frac{\frac{1}{N} \sum_{i=1}^N P(C_k / \underline{X}_i)}{\sum_k \left(\frac{1}{N} \sum_{i=1}^N P(C_k / \underline{X}_i) \right)} \quad 3-3$$

Die Stärke des Umgebungseinflusses wird über eine Konstante festgelegt, den Grad der Schärfe $GdS \in \mathfrak{R}[0,1]$. Je höher der Grad der Schärfe gewählt wird, desto stärker ist die Klassenentscheidung vom Prior, d.h. von der lokalen Nachbarschaft, beeinflusst, woraus ein Anstieg des Glättungseffektes folgt. Bei Wahl von $GdS=0$ resultiert die Klassenentscheidung aus der alleinigen Auswertung der Likelihoodfunktion $P(\underline{X}_0/C_k)$. Die Klassenzuordnung mit Prior an der Stelle des Aufpunktes P_0 im Labelbild erfolgt entsprechend dem maximalen Posterior $P(C_k/\underline{X}_0)$ nach Formel 3-4.

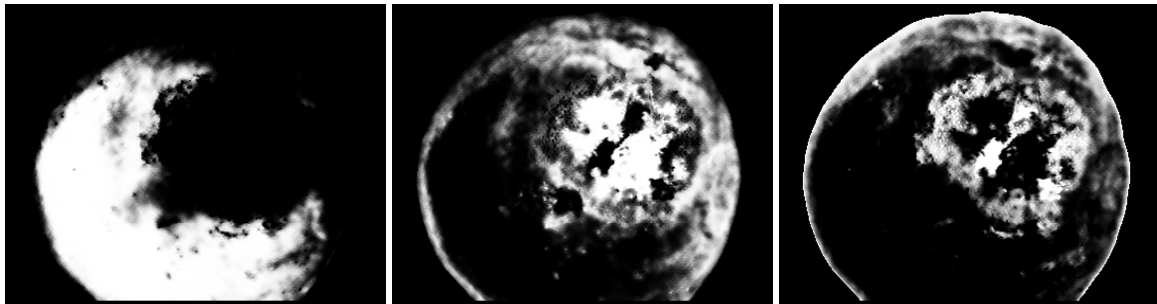
$$\arg \max_{C_k} \{P(C_k / X_0)\} \propto \arg \max_{C_k} \left\{ w(\underline{X}_0 / C_k)^{1-GdS} \cdot P(C_k)_{|U(P_0)}^{GdS} \right\} \quad GdS \in \mathfrak{R}[0,1] \quad 3-4$$

Die Bayessche Filterung wurde auf Basis der Membershipbilder durchgeführt. In der folgenden Abbildung ist rechts das Ergebnis der Bayesschen Filterung mit einer Filtergröße von $N = 7 \times 7 - 1$ und einem Grad der Schärfe von 0.8 im Vergleich zum ursprünglichen Labelbild (links) zu sehen.



Unter Nutzung dieses bayes-gefilterten Labelbildes ist es möglich, Objekte (Schadstellen) im Anschluss durch eine schnelle Segmentierung mittels Zeilenkoinzidenz zur weiteren Entscheidungsfindung zu erfassen.

Alternativ können nach einer Normierung von $w(\underline{X}_0 / C_k)^{1-GdS} \cdot P(C_k)_{|U(P_0)}^{GdS}$ für alle Klassen k Membershipbilder ausgegeben werden, welche die Posteriori-Wahrscheinlichkeiten abbilden. Die Segmentierung auf Basis der bayes-gefilterten Membershipbilder ist ein weiterer Ansatz zur Objektdetektion, der in Kapitel 4.1.2 vorgestellt wird.



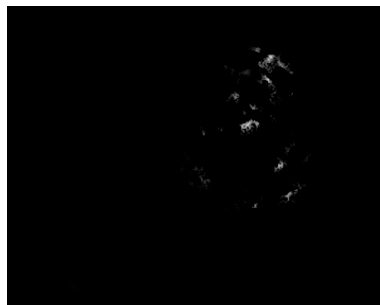
Bayes-Membershipbild Label 1 (gut)

Bayes-Membershipbild Label 2 (weißer Schimmel)

Bayes-Membershipbild Label 3 (schwarzer Schimmel)



Bayes-Membershipbild Label 4 (grün)



Bayes-Membershipbild Label 5 (Fäule)

4 Globale und regional integrale Merkmale zur Realisierung von Güteklassen für die Gutobjekte - symbolische Klassifikation

4.1 Segmentierung unter Verwendung von Klassenzugehörigkeitsmaßen

4.1.1 Motivation

Mit der Klassifikation auf Basis rein globaler Merkmale (Flächenanteile der Label im Bild) aus den unstrukturierten Bildinhalten lassen sich keine zufrieden stellenden Zuordnungen zu einer symbolischen Klasse treffen. Aus globaler Sicht können z.B. zusammenhängende, weiße Schadstellen (weißer Schimmel) nicht von einer Vielzahl von weißen Einzelpixeln (Reflexe auf der gewachsenen Orangenhaut) unterschieden werden, obwohl nur erstere zu Abfallorangen führen müssen. Die abschließende Klassifikation in Abfall und gute Früchte mit Unterklassen zum Degreening und zur Bedienung unterschiedlicher Marktsegmente erfordert also die Berechnung lateral zusammenhängender Segmente und, je nach Einsatzfall, die Berechnung unterschiedlicher Segmentmerkmale.

Basis für die Segmentierung sind die Ergebnisse der Pixelklassifikation, d.h. Klassenlabel der harten, ikonischen Klassifikation oder der Bayesschen Filterung lt. Abschnitt 3.4.2.

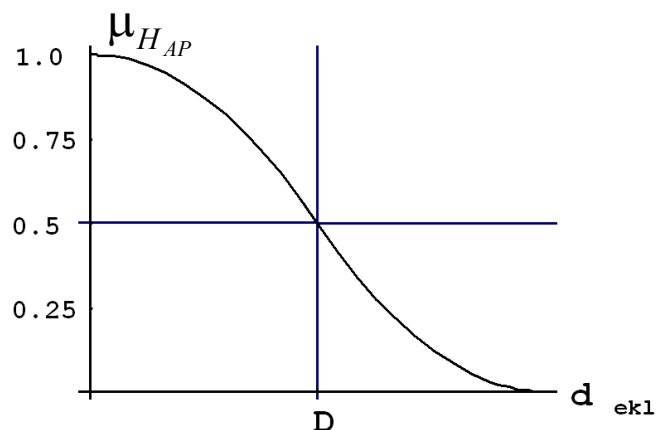
Um die Kette der unscharfen Verarbeitung weiterzuführen und Quantisierungsfehler aus einer scharfen Segmentierung zu vermeiden, wurde ein unscharfes Segmentierungsverfahren entwickelt, dass auf dem Region-Growing-Ansatz basiert.

4.1.2 Fuzzy-Region-Growing

Im Gegensatz zum quasiparallelen Wachstum einer Vielzahl von Saatpunkten und Verschmelzen ähnlicher Regionen nach Abschluss der Segmentierung lässt man beim abgewandelten unscharfen Algorithmus immer nur einen Saatpunkt bis zum vollständigen Teilsegment auswachsen. Das ist möglich, da sich die Segmente beim Wachstum in den bayes-gefilterten Membershipbildern in keiner Weise gegenseitig beeinflussen und sich dadurch natürliche unscharfe Segmentgrenzen bilden können.

Zur Saatpunktpositionierung bietet sich die Verwendung des Minimums im Streuungsbild der Eingangskanäle (hier bayes-gefilterte Membershipbilder), ausgehend vom zuletzt gefundenen Segment. Dadurch wird abgesichert, dass mit dem Segmentwachstum in einem möglichst homogenen Bereich begonnen wird.

Für jedes potentiell anzulagernde Pixel AP aus der Nachbarschaft des Saatpunktes oder eines Randpixels des wachsenden Segmentes wird ein Wert zwischen 0 und 1 ermittelt, der den Erfüllungsgrad des Homogenitätskriteriums widerspiegelt. Als Distanzmaß wird die euklidische Entfernung d_{ekl} des Pixelmerkmalsvektors zum Segmentmerkmalsvektor benutzt. Zur Bestimmung des Erfüllungsgrades findet eine Z-Zugehörigkeitsfunktion mit parametrierbarer „Bandbreite“ D Verwendung.



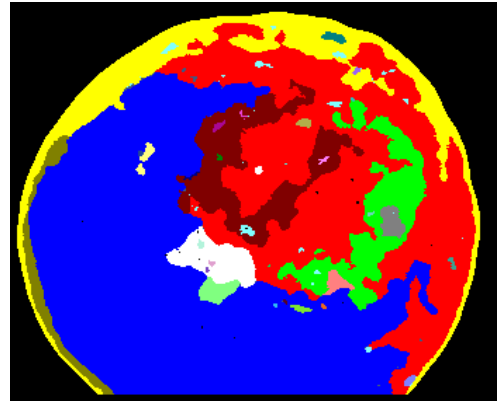
Die Angliederung eines Pixels AP an ein Segment erfolgt über eine Verknüpfung mit seinem Ursprungsrangepixel UP . Für AP wird das Pixel ausgewählt, welches die größte Segmentzugehörigkeit von allen benachbarten Randpixeln aufweist. Die Segmentzugehörigkeit für das anzulagernde Pixel AP wird dann über die so genannte Fuzzy-Verbundenheit (fuzzy-connectedness) wie folgt bestimmt:

$$\mu_{S_{AP}} = \min \left(\mu_{H_{AP}}, \arg \max_{UP} (\mu_{S_{UP}}) \right) \quad 4-1$$

Eine Region gilt als vollständig ausgewachsen, wenn, ausgehend von deren Randpixeln, keine benachbarten Pixel mit einem Erfüllungsgrad der Homogenitätsbedingung von $\mu_{H_{AP}} > 0$ mehr vorkommen. Der Algorithmus endet, wenn alle unmaskierten Bereiche segmentiert wurden.

Nach Abschluss der Segmentierung werden die Zugehörigkeiten der Pixel normiert. Jedes Pixel ist dann vollständig auf die Segmente aufgeteilt und kann gleichzeitig mehreren Segmenten angehören, jeweils zu einem bestimmten Grad.

Für das Beispiel „Orangeninspektion“ wurde als Bandbreite für die Zugehörigkeitsfunktion der Wert 80 eingestellt. Als Eingangsbilder wurden die fünf bayes-gefilterten Membershipbilder aus Kapitel 3.4.2 verwendet. Nur zur Darstellung ist rechts für jedes Pixel das Segment mit der größten Zugehörigkeit dargestellt. Für die Berechnung von Segmentmerkmalen wird der Vorteil der unscharfe Segmentzuordnung genutzt.



4.2 Ableitung von Regionenmerkmalen und symbolischer Klassifikator

Der symbolische Klassifikator bildet die letzte Stufe der Verarbeitungskette im Inspektionsprozess.

Für jede der Regionen können, je nach vorangegangener Segmentierungsvariante (scharf oder unscharf), geometrische bzw. fuzzy-geometrische Merkmale berechnet werden. Als Oberflächenmerkmale für Schadstellen auf Orangen oder für den Reifegrad sind die Flächen der entsprechenden Segmente oder Flächenverhältnisse von Bedeutung. Für Schadstellen, die aus mechanischen Beschädigungen resultieren, kommen Formfaktoren hinzu. Solche Formmerkmale und Merkmale aus Farbtexturen spielen im Zusammenhang mit vielen Schüttgütern aus dem Bereich der Lebensmittel- und Grundstoffindustrie, die durch den Ansatz ebenfalls erschlossen werden, oft eine starke Rolle.

Die unscharfe Klassifikation über Fuzzy-Inferenz-Systeme eignet sich zur Bewertung von (fuzzy-) geometrischen Segmenteigenschaften. Die Ergebnisse werden mit den Attributen der zugehörigen Pixel (z.B. mittlere Farbvalenzen bzw. Klassen der ikonischen Klassifikation) zu Regionenmerkmalen verknüpft, die für die endgültige symbolische Klassifikation geeignet sind. So entstehen beispielsweise in der Applikation Orangeninspektion die neuen Klassen „Schimmel weiß kleinflächig“ und „Schimmel weiß großflächig“. Auf diese Unterklassen werden die Anteile der „kleinflächigen“ bzw. der „großflächigen“ Regionen verteilt, die wiederum mit der Zugehörigkeit des Segments zum ikonischen Merkmal „weißer Schimmel“ gewichtet werden.

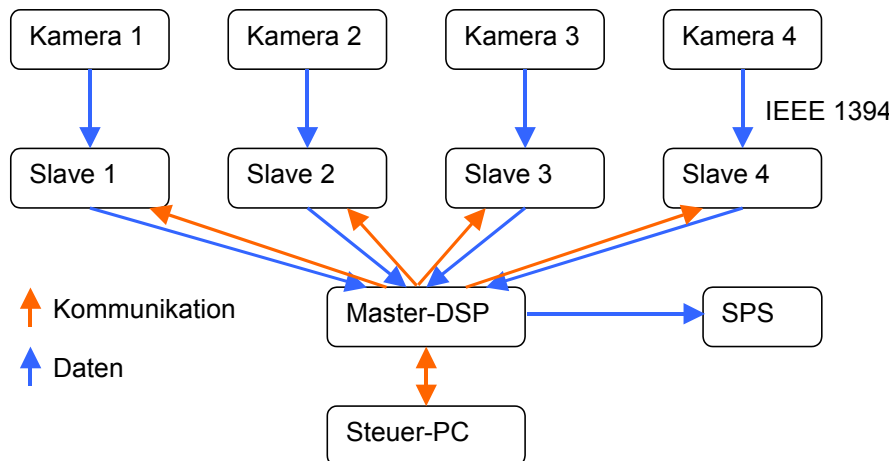
Auf Grundlage dieser globalen Merkmale wird ein symbolischer Klassifikator mit einer ausreichend großen Stichprobe von Repräsentanten der endgültig zu unterscheidenden Klassen (z. B. Abfall und Gutklassen mit Unterklassen wie Degreening bzw. Güteklassen für den Markt) trainiert. Es sei an dieser Stelle erwähnt, dass oft weitere Objektmerkmale verwendet werden, die nicht aus Bilddaten gewonnen werden. Sie lassen sich aber im Allgemeinen problemlos in das angegebene Konzept integrieren. Im Falle der Orangen ist das z.B. das Gewicht, dass zusammen mit der Größe zur Schätzung des Saftgehaltes verwendet wird.

5 Aspekte der rechentechnischen Umsetzung, Echtzeit und Parallelität

5.1 Hardwarestruktur der Verarbeitungseinheit, Aufteilung der Algorithmen und Performance

5.1.1 Hardwarestruktur

Die Hardwarestruktur der Verarbeitungseinheit wird durch die kameratechnischen, algorithmischen und zeitlichen Erfordernisse bestimmt. Aus dem geforderten Durchsatz und der gewünschten Robustheit des



Systems lässt sich sowohl für die Orangeninspektion als auch für das Sortieren von Schüttgütern im Allgemeinen ableiten, dass der potentielle Datendurchsatz und die Rechenleistung eines PCs nicht ausreichen. Die Aufgabe wurde deshalb mit einem Mehrfach-DSP-Systems gelöst.

Jeder Kamera ist ein Slave-DSP zugeordnet, der die ikonische Bildverarbeitung übernimmt. Die Slave-DSPs

senden ihre Ergebnisse über eine High-Speed-Verbindung an einen Master-DSP, der die Endklassifikation übernimmt. Über digitale Schnittstellen können Sortierweichen direkt angesteuert werden. Der Master-DSP übernimmt außerdem die Betriebsarten-Steuerung des DSP-Systems und die Kommunikation mit dem Steuer-PC über Ethernet.

Die Verwendung eines Steuer-PC ist mit einer Reihe von Vorteilen verbunden. Einerseits wurde die Basislösung auf PC-Grundlage entwickelt. In der gewählten Struktur müssen nur die in der Ausführungsphase benötigten Teilalgorithmen auf die DSPs portiert werden. Für alle anderen Prozesse werden die vielfältigen Interaktionsmöglichkeiten und peripheren Einheiten des PC genauso genutzt wie die Netzanbindung und allgemein genutzte Programme (Excel, Access, ...).

Zu diesen auf dem PC laufenden Prozessen gehört unter anderem das Anlernen der verschiedenen Klassifikatoren. Das DSP-System verfügt über unterschiedliche Betriebsarten, um diese Vorgehensweise zu unterstützen. Im Bildtransfer-Modus werden die Kamerabilder nur aufgenommen und an den PC übertragen, wo sie chargenweise unterteilt nach Fehlerklassen gespeichert werden. Auf dem PC kann der Bediener fehlerrelevante Regionen markieren und die Klassifikatoren anlernen. Mit den gespeicherten Bildern wird anschließend eine Reklassifikation durchgeführt, um die Belehrungsqualität einschätzen zu können.

Nach dem erfolgreichen Belehren werden die Dateien mit den Parametersätzen auf die DSPs übertragen und die Echtzeit-Pipelines initialisiert. Neben dem Echtzeitmodus wurden verschiedene Testmodi vorgesehen, welche die Bewertung einzelner Früchte durch erweiterte Anzeigefunktionen und Ausgaben gestatten.

5.1.2 Performance

Auf einem Standard-PC benötigt die Verarbeitung einer Kameraansicht ca. 80 ms. Für alle Ansichten und die Zusammenfassung der Teilergebnisse werden insgesamt 330 ms benötigt, was ca. 3 Früchten pro Sekunde entspricht.

Die DSPs benötigen dagegen jeweils 13 ms für eine Kameraansicht und die Zeiten addieren sich wegen der Parallelverarbeitung nicht. Ihre Hintereinanderschaltung (zeitlich versetzte Einzelaufnahmen zur Verhinde-

rung von Blendeffekten) hat nur Einfluss auf die Durchlaufzeit der Daten, aber nicht auf die Taktrate. Insgesamt können bis zu 60 Objekte pro Sekunde inspiziert werden. Der tatsächliche Durchsatz wird für die meisten Anwendungen durch die Fallstrecke und die mechanischen Randbedingungen der Maschinenlösung limitiert.

Es ist offensichtlich, dass die Orangeninspektion zu den Einsatzfällen gehört, die weit unter dem dargestellten Maximaldurchsatz pro Sortierkanal bleiben. Die Zeitreserven können hier zu Gunsten hoher Inspektionsqualität genutzt werden, um komplexe Algorithmen im vorangehend diskutierten Sinne zu integrieren. Bei den bereits erwähnten Schüttgütern wird der realisierte Durchsatz der beschriebenen Lösung benötigt.

5.2 Signalprozessoren und Aspekte der Portierung

Bei der Portierung der Algorithmen musste darauf geachtet werden, dass auf den DSPs bis ins Detail die gleichen Ergebnisse erzielt werden wie auf dem PC, damit der PC zum Anlernen und zur Bewertung von chargen- bzw. jahresabhängigen Parametrierungen genutzt werden kann. Der Ersatz von Floating-Point-Berechnungen auf dem PC durch genäherte Fix-Point-Berechnungen auf dem DSP führte zu kleineren Abweichungen, die gerade noch tolerierbar waren.

Der DSP kann auf verschiedenen Abstraktionsebenen programmiert werden, beginnend bei C++ über C bis hin zum Assembler. Die Wahl richtet sich nach den Anforderungen an die Komplexität und die Zeiteffizienz. In jedem Fall übernimmt der Compiler die Optimierung. Für das Steuerprogramm auf dem Master-DSP reicht vereinfachtes Programmieren in C++.

Um kurze Verarbeitungszeiten für die Bilddaten auf den Slave-DSPs zu erzielen, ist es erforderlich, deren Potential zur Parallelverarbeitung nach dem SIMD-Prinzip auszuschöpfen. Das ist mit Standard-C-Quelltext nicht erreichbar. Deshalb wurden einerseits in C-Modulen die vom Compiler angebotenen Pseudofunktionen (Intrinsics) mit speziellen Datentypen benutzt und andererseits die Programmierung der zeitkritischen Bausteine direkt in Assembler vorgenommen.

Schließlich mussten die Algorithmen in Bezug auf die Speicherorganisation optimiert werden. Die Bildspeicher wurden so strukturiert, dass vorwiegend auf dem sehr schnellen internen RAM des DSP gearbeitet werden kann und Datentransfers zwischen internem und externem RAM per DMA erfolgen.

Danksagung

Den Kollegen **Prof. Giannis Kalikatsos**, **Prof. George-Othon Glentis** und **Prof. Emmanuel Antonidakis** vom Technological Educational Institute of Crete, Branch of Chania (Griechenland), der Landwirtschaftlichen Genossenschaft MALEME Co-op (Kreta) sowie den Studenten **Ch. Karl**, **S. Rührschneck**, **Th. Leucht**, **D. Güllmar** sei recht herzlich für die aktive Unterstützung, den Fleiß bei der Zusammenstellung der Stichproben und für die Durchführung vielzähliger Experimente gedankt. Dank auch an Herrn **Privatdozent Dr. P. Zocher** von der Technischen Universität Ilmenau, der im Rahmen des Programmes ERASMUS/SOKRATES die Vorarbeit zur Aufgabenstellung geleistet und das Projekt kontinuierlich unterstützt hat.

6 Literatur

- /ZBS06/ ZBS e.V., R. Nestler: „ClusterLib – Bibliothek“,
<http://www.zbs-ilmenau.de/html/prod/cluster.html>,
http://www.zbs-ilmenau.de/pdf/Flyer_22_Clusterbibliothek.pdf , Ilmenau 2006
- /Fra95/ K.-H. Franke: "Nutzung von Farbinformation zur Fehlerdetektion und Klassifikation von Fehlern auf strukturierten Halbleiterwafern", in Tagungsband zum 1. Workshop Farbbildverarbeitung, Fachberichte Informatik 15/95, Universität Koblenz-Landau, 1995
- /Fra96/ K.-H. Franke, T. Machleidt: "Farbpixelklassifikation mittels partitionierender Clusteralgorithmen und Vergleich mit neuronalen Ansätzen", in Tagungsband zum 2. Nationalen Workshop Farbbildverarbeitung, 10.-11.10.1996, Schriftenreihe des ZBS e.V., ISSN 1432-3346, S. 61-64