

Inhaltsverzeichnis

Sitzung ABC

ECCi: Neuer 2D-Matrixcode für sicheres Lesen von direkt beschrifteten Industrieprodukten	1
<i>H. Tropsch (Vision Tools Bildanalyse Systeme GmbH)</i>	

ECCi: Neuer 2D-Matrixcode für sicheres Lesen von direkt beschrifteten Industrieprodukten

Dr.-Ing. Hermann Tropf

Vision Tools Bildanalyse Systeme GmbH,
Goethestraße 65, D-68789 Waghäusel

Zusammenfassung Industrieprodukte werden vielfach mit Matrix Codes gekennzeichnet zur Organisation der Logistik oder zur Sicherstellung der Produkt-Rückverfolgbarkeit. Bei dem etablierten „Data Matrix Code“ (ECC 200, ECC für „Error Correcting Code“) treten jedoch bei Direktbeschriftung trotz Normungsbemühungen und trotz erheblicher Fortschritte bei der Bildauswertungssoftware immer wieder Probleme mit der Sicherstellung der Lesbarkeit auf. Der Grund liegt im komplizierten Zusammenwirken von Beleuchtungs- und Bildaufnahme-Geometrie, Beschriftungsart, Beschriftungsparametern, Oberflächenform und Oberflächencharakteristik, das bei der Projektierung einer Anwendung zu berücksichtigen ist.

Die ECCi-Codierung/Decodierung (ECCi: Error Correcting Code industrial) arbeitet differentiell, nichtiterativ und mit Soft-Decision. Aufgrund der differentiellen Arbeitsweise ist der Code lesbar unabhängig von der optischen Erscheinungsform der Zellen und somit unabhängig von der Beleuchtungs- und Beschriftungsweise.

1 Einführung

In der industriellen Produktion werden Produkte mit Matrix Codes gekennzeichnet zur Organisation der Logistik oder zur Sicherstellung der Produkt-Rückverfolgbarkeit. Der „Data Matrix Code“ (die etablierte Version ist ECC 200, ECC steht für „Error Correcting Code“) [1] wurde entwickelt für auf Papier beschriftete Vorlagen als Ersatz von eindimen-

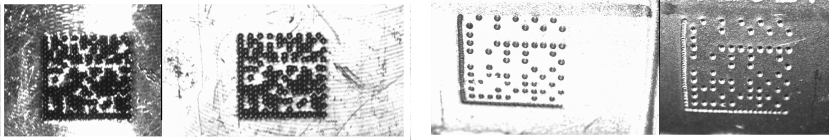
sionalen Barcodes, gegenüber denen sie eine weitaus höhere Datendichte aufweisen.

Ein Matrixcode allgemein besteht aus einer matrixförmigen Anordnung von informationstragenden, binär codierten Zellen (schwarz/weiß; bearbeitet/unbearbeitet, kurz ON-/OFF-Zellen), ergänzt durch einen Locator, mit dessen Hilfe die genaue Position der informationstragenden Zellen innerhalb des Codes bestimmt wird. Der Locator besteht bei ECC 200 aus zwei Balken in L-förmiger Anordnung, und gegenüberliegend aus den sog. Frequenzlinien (s. Abb. 1.1(a) und 1.1(b)).

Bei auf Etiketten gedruckten Codes leistet ECC 200 schon lange Zeit hervorragende Dienste. Der Code wird zunehmend auch für Anwendungen mit Direktbeschriftung (Laserbeschriftung, Nadelbeschriftung) übernommen. Das Lesen ist bei Direktbeschriftung i.a. schwieriger als bei Etiketten, und trotz erheblicher Fortschritte bei der Bildauswertungssoftware treten hierbei immer wieder Probleme mit der Sicherstellung der Lesbarkeit auf. Der Grund liegt im komplizierten Zusammenwirken von Beleuchtungs- und Bildaufnahme-Geometrie, Beschriftungsart, Beschriftungsparametern, Oberflächenform (eben, gekrümmt...) und Oberflächencharakteristik (matt, spiegelnd, zerklüftet), das bei der Projektierung einer Anwendung zu berücksichtigen ist. Es existieren verschiedene Normen, mit denen versucht wird, die Qualität der Beschriftung sicherzustellen [2, 3]; es gibt auch käuflich erhältliche sog. Verifier, die auf Basis dieser Normen die Qualität der Markierung sicherstellen sollen. Die Sicherstellung der Lesbarkeit jedoch gelingt damit für viele Industrieanwendungen mit Direktmarkierung nicht, denn je nach Situation ist für garantiert sicheres Lesen eine unterschiedliche Beleuchtungs- und Aufnahmegeometrie optimal oder gar notwendig; sehr oft sogar weicht diese von der Norm ab.

Hierzu einfache Beispiele: Wie aus Abb. 1.1(a) ersichtlich, kann bei glänzenden Oberflächen die Darstellung kritisch abhängig sein von der Größe der beleuchtenden Fläche. Abb. 1.1(b) zeigt, wie je nach Beleuchtungsgeometrie eine Kontrastumkehr entstehen kann. Speziell bei Gussteilen kann je nach Beleuchtung die Hervorhebung der Matrix gegen den Hintergrund sehr kritisch sein. Eine ausführliche Beschreibung mit Fallunterscheidungen zu Oberflächenform, Oberflächen-Struktur und Beschriftungsart findet man in [4].

Der Einsatz eines Verifiers ist zur Lesbarkeit nur dann sinnvoll, wenn der Aufbau des Verifiers exakt dem der nachfolgenden Lesestationen ent-



(a) Abhängigkeit von der Größe der beleuchtenden Fläche

(b) Hellfeld vs. Dunkelfeld

Abbildung 1.1: Beleuchtungsabhängigkeit der optischen Darstellung

spricht. Ist dies nicht der Fall, kann der Einsatz eines solchen Systems sogar kontraproduktiv sein in dem Sinne, dass gerade Beschriftungsparameter, für die der Verifier optimale Ergebnisse liefert, im späteren Einsatz zu schwierig/nicht lesbaren Vorlagen führen.

Ein weiteres Problem bei ECC 200 ist der Locator, mit dessen Hilfe der Code im Bild gesucht wird: Um den Locator sicher auswerten zu können, wird eine sog. Quiet Zone von mindestens einer Zellgröße um die Matrix herum gefordert, was bei strukturierten Oberflächen oder bei Platzproblemen nicht immer eingehalten werden kann. Bei gestörtem Locator kann der Code nicht gefunden und somit auch nicht gelesen werden.

ECC 200 basiert auf Reed-Solomon (RS) Codierung/Decodierung mit Blöcken von 8-Bit-Symbolen [1]. Die gängigen Decodieralgorithmen für ECC 200 arbeiten rein algebraisch mit sog. Hard-Decision. Günstiger wäre Soft-Decision-Decodierung, wie aus der Kanalkodierung bekannt. Soft-Decision-Decodierer verwenden Zuverlässigkeitsinformation, die mit den Eingangssymbolen verbunden ist, und liefern daher grundsätzlich bessere Ergebnisse als entsprechende Hard-Decision-Decodierer, die von Zuverlässigkeitsinformation keinen Gebrauch machen.

In neuerer Zeit wurden die iterativ arbeitenden Turbo-Codes erfolgreich eingesetzt. Turbo-Codes können in bestimmten Fällen erstaunlich stark gestörte Signale decodieren, jedoch führt die iterative Arbeitsweise zu nicht vorhersagbarer Rechenzeit; die Wirkungsweise ist schwierig zu durchschauen, der Entwurf und die Optimierung für eine spezielle Anwendung sind schwierig. Beliebte sind derzeit sog. Low-Density-Parity-Check-Codes (LDPC). Die Decodierung ist ebenso iterativ; Soft-Decision ist möglich. Eine Vergleichsstudie von LDPC mit gängiger Reed-Solomon-Decodierung, angewendet auf direkt beschriftete Matrixcodes, wurde kürzlich erstellt und verweist auf eine höhere Störsicherheit bei

LDPC [5].

Die im folgenden in Grundzügen beschriebene ECCi-Codierung/Decodierung (ECCi: Error Correcting Code industrial) arbeitet differentiell (s.u.), nichtiterativ und mit Soft-Decision. Ein Vergleich mit LDPC steht noch aus.

2 Grundzüge der neuen Codierung/Decodierung

2.1 Differentielle Codierung

Die Codierung ist im Gegensatz zu ECC 200 differentiell. Grundoperation beim Decodieren ist der Vergleich von Zellinhalten (kleine Bildvergleiche). Dem liegt die Erkenntnis zugrunde, dass, unabhängig von der Beschriftungsweise und unabhängig von der Beleuchtungsweise, gleichartige Zellen (also 2 ON-Zellen bzw. 2 OFF-Zellen) sich immer gleichartig darstellen. Dies gilt zumindest innerhalb einer lokalen Umgebung, siehe Bildbeispiel in Abb. 1.2.

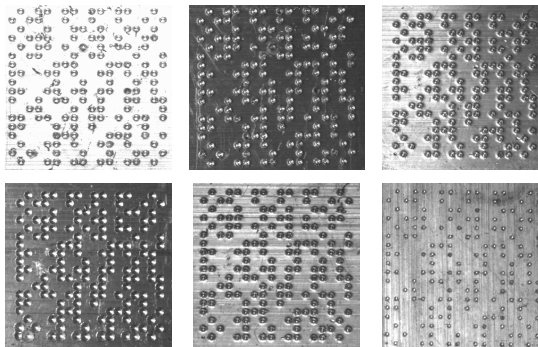


Abbildung 1.2: ON- bzw. OFF-Zellen erscheinen zumindest lokal unabhängig von Beleuchtung und Beschriftungsweise immer gleichartig

Damit ist der Code lesbar unabhängig von der optischen Erscheinungsform der Zelle und somit unabhängig von der Beleuchtungs- und Beschriftungsweise. Dieser Vorteil ist besonders frappant bei verformender Markierung: Reflex- und Schatteneffekte sind bei gleichartig behandelten Zellen lokal immer ähnlich, das gilt auch bei einer sich zufällig ergebenden Kontrastumkehr.

2.2 Soft-Decision und Mehrheitsentscheidungen

Beim Decodieren werden dementsprechend jeweils zwei Zellen in nicht allzu großem Ortsabstand auf Gleichheit/Ungleichheit untersucht. Die Entscheidungen „gleich oder ungleich?“ werden bewertet (z.B. „sicher gleich... eher gleich... unsicher... eher ungleich... sicher ungleich“); jeweils mehrere solcher Vergleiche werden Mehrheitsentscheidungen unterworfen. Dadurch werden lokale Störungen überwunden. Die Decodierung ist so organisiert, dass jede Einzelentscheidung auf einer Mehrzahl von Zellvergleichen beruht, mit einer nachgeschalteten Mehrheitsentscheidung. Man kann das am besten mit einer Meinungsbildung von mehreren Personen vergleichen, wobei jede Person entsprechend der (subjektiven) Sicherheit ihrer Meinung ihr persönliches Gewicht in die Waagschale werfen und sich ggf. auch der Stimme enthalten kann. Kleinere lokale Störungen werden auf diese Weise eliminiert.

Anschließend werden die Bewertungen einem global über das Codemuster sich erstreckenden Soft-Decision-Optimierungsverfahren unterworfen. So werden zusätzlich größere Störungen überwunden.

2.3 Integrierter Locator

ECCi ist, wie technisch realisiert, selbstsynchronisierend, grundsätzlich kann daher auf einen speziellen Locator verzichtet werden. So ist für Codes, die beliebig verschoben und in grob bekannter Drehlage (± 10 Grad) präsentiert werden, kein Locator erforderlich. Zur Beschleunigung der Lokalisierung werden allerdings für Anwendungen, bei denen zusätzlich die Drehlage gänzlich unbekannt ist, an verschiedenen Stellen kleine schachbrettartige Muster eingestreut [6]. Vorteil solcher eingestreuter Muster ist, dass - im Gegensatz zu ECC 200 - Störungen am Rand die Lokalisierung nicht beeinträchtigen. Es wird keine Quiet Zone gefordert, es kann bis zum Rand und sogar etwas darüber hinaus beschriftet werden. Aber auch lokale Störungen innerhalb der Matrix beeinträchtigen aufgrund eingebauter Redundanz die Lokalisierung nicht.

3 ECCi-Codierung und -Decodierung

3.1 Ein Beispiel

Hier wird der Kern-Prozess der Codierung sowie der differentiellen Decodierung mit Mehrheitsentscheidungen erläutert - anhand eines einfachen, aber instruktiven Beispiels, mit zunächst binär interpretierten Zellen. Danach wird der Übergang auf Zuverlässigkeitsinformation kurz erläutert. Zu Details dazu und weitergehenden Maßnahmen, wie der Realisierung von zellweisen Bildvergleichen und Mehrheitsentscheidungen, muss aus Platzgründen auf das Patent [7] verwiesen werden.

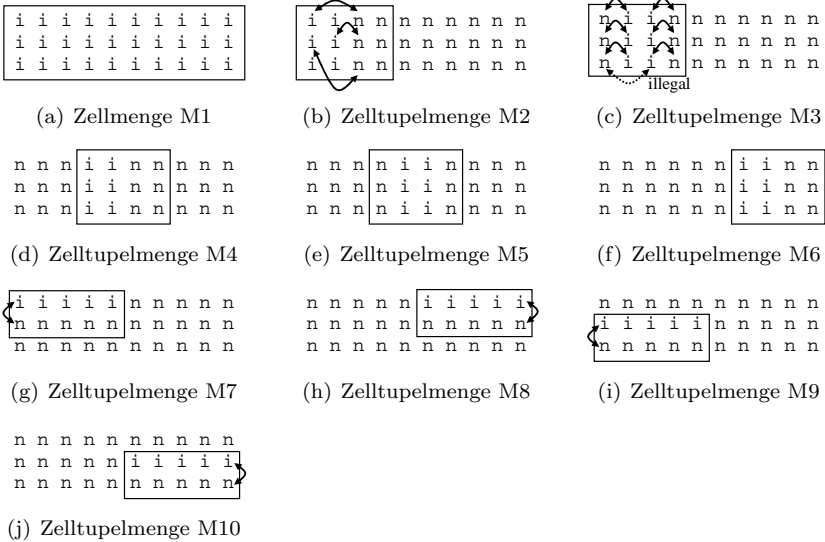


Abbildung 1.3: Zum Beispiel ECCi-Codierung/Decodierung

Im Beispiel Abb. 1.3(a) bis 1.3(j) soll eine 10-Bit-Information codiert werden. Gegeben sei ein Feld mit 10×3 Zellen. Die 30 Zellen sollen redundant für eine Codierung der 10 Infobits verwendet werden. Die Anordnung der Zellen muss nicht unbedingt der endgültigen Anordnung entsprechen, weil am Ende noch eine gestreute Platzierung, ähnlich Interleaving, durchgeführt wird (s.u.). Es handelt sich um eine sequentielle

Codierung/Decodierung, infobitweise, im Beispiel also mit 10 Schritten.

3.2 ECCi-Codierung anhand Beispiel

Als Initialisierung wird allen Zellen des zu erstellenden Codes der gleiche Grundwert, z.B. 0, zugewiesen.

Für den ersten Codierschritt (erstes Infobit) definieren wir eine Menge M_1 von Zellen, die in unserem Beispiel speziell alle Zellen umfaßt. Die Zellen in M_1 werden nun alle invertiert, wenn das erste Infobit 1 ist; falls es 0 ist werden sie nicht invertiert. Mit anderen Worten: die Zellen in M_1 werden XOR-verknüpft mit dem ersten Infobit. Solcherart behandelte Zellen von Zellmengen werden im folgenden Inversionszellen genannt und in den Abb. 1.3(a) bis 1.3(j) mit „i“ dargestellt. Nichtinversionszellen, mit „n“ dargestellt, bleiben dagegen unabhängig vom Infobit unverändert.

Für die folgenden Infobits definieren wir je eine Menge von Zell tupeln. Ein Zelltupel besteht aus mindestens einer Inversionszelle und mindestens einer Nichtinversionszelle. Da es insgesamt 10 Infobits zu codieren gilt, definieren wir noch insgesamt $10 - 1 = 9$ Zell tupelmengen.

In unserem Beispiel besteht jedes Zelltupel aus genau einer Inversionszelle und einer Nichtinversionszelle. Einige Zell tupel sind in Abb. 1.3 als Doppelpfeil zwischen je einer Inversionszelle und einer Nichtinversionszelle angedeutet.

Bei der Definition der Zellpaare muss folgende Vorschrift beachtet werden: Die Zellen eines Zell tupels müssen bei den XOR-Operationen aller vorangehenden Schritte jeweils alle gleich behandelt worden sein, unabhängig vom aktuellen Wert der vorangehenden Infobits. Dies ist erfüllt, wenn sie für jeden vorangehenden Codierschritt jeweils entweder alle Inversionszellen oder alle Nichtinversionszellen sind.

In Abb. 1.3(b) umfaßt die Zell tupelmenge M_2 6 Inversionszellen und 6 Nichtinversionszellen. Es sind Überlappungen erlaubt, d.h. eine Zelle kann mehreren Zell tupeln angehören; lediglich die oben aufgeführte Vorschrift muss beachtet werden.

Die Inversionszellen von M_2 werden mit dem zweiten Infobit XOR-verknüpft. D.h. falls sie aufgrund des ersten Infobits bereits invertiert wurden und nun aufgrund des zweiten Infobits invertiert werden, nehmen sie wieder den ursprünglichen Grundwert an.

Beim dritten Infobit mit Zell tupelmenge M_3 , siehe Fig. 1.3(c), umfasst die Menge M_3 in unserem Beispiel genau die gleichen Zellen wie M_2 .

Ein im Sinne der oben aufgeführten Vorschrift illegales Zelltuplel ist in Abb. 1.3(c) angegeben (gestrichelt). Es ist deshalb illegal, weil seine Nichtinversionszelle abhängig vom zweiten Infobit invertiert wurde, seine Inversionszelle hingegen nicht. Bei Infobit 4 (M4, Abb. 1.3(d)), sind die Zellpaare wie vorher bei M2 gewählt. Bei Infobit 5 (M5, Abb. 1.3(e)), sind die Zellpaare wie vorher bei M3 gewählt. Bei Infobit 6 (M6, Abb. 1.3(f)), sind die Zellpaare wie vorher bei M2 gewählt. Bei den Infobits 7 bis 10, mit den Zelltuplelmengen M7 bis M10, siehe Abb. 1.3(g) bis 1.3(j) ist in diesem speziellen Beispiel die obige Vorschrift immer dann erfüllt, wenn die Zellen eines Zellpaares übereinander stehen.

Die schrittsspezifische Aufteilung in Zelltuplelmengen, Aufteilung in Zelltuplel und Zuweisung der Inversions- und Nichtinversionszellen eines Zelltuplels, Konfiguration genannt, geschieht nur einmal vorab, manuell oder mit Rechenprogramm. Die Codierung besteht einfach in der Anwendung der XOR-Operationen auf die Inversionszellen entsprechend der Konfiguration, infobitweise.

3.3 ECCi-Zellplatzierung

Da geometrisch gesehen der Platz, wo die Zellen eines Zellpaares sitzen, keine Rolle spielt, sondern nur die Zuordnung zu den Mengen, haben wir noch die Freiheit, die Zellen beliebig zu platzieren. Wir führen nun noch eine gestreute Platzierung durch, derart dass einerseits Zellen desselben Zelltuplels möglichst nah beisammen liegen und andererseits Zelltuplel, die über Zelltuplelmengen miteinander verbunden sind, möglichst weit voneinander platziert werden. Der Effekt ist ähnlich dem bekannten Interleaving, durch das nicht nur Einzelfehler, sondern auch Bündelfehler korrigiert werden können. Die Platzierungsvorschrift muss natürlich dem Codierer wie dem Decodierer bekannt sein.

3.4 ECCi-Decodierung anhand Beispiel

Die Decodierung basiert auf dem Vergleich des Inhalts von Inversionszellen und Nichtinversionszellen innerhalb von Zelltupleln. Die Zelltuplelmengen und damit die Teilinformationen werden gegenüber dem Codieren in umgekehrter Reihenfolge bearbeitet, hier also M10, M9, ..., M1.

Da wir im Beispiel binär codierte Zellen haben und da die Zelltuplel hier Zellpaare mit je einer Inversionszelle und einer Nichtinversionszelle

sind, ist die Vorgehensweise hier wie folgt:

Sind die jeweils betrachteten Zellen eines Zellpaares ungleich, so spricht das dafür, dass das zugehörige aktuelle Infobit gesetzt ist; sind sie gleich, so spricht das dafür, dass das zugehörige aktuelle Infobit nicht gesetzt ist. Die Entscheidung wird vorzugsweise über einen Mehrheitsentscheid der beteiligten Zellpaare einer Zelltupelmenge getroffen.

Sofern weitere Decodierschritte anstehen, werden jedesmal, wenn eine Entscheidung daraufhin gefällt wurde, dass das Infobit gesetzt wird, die Inversionszellen der aktuellen Zellmenge invertiert. Auf diese Weise werden also nacheinander die Infobits 10, 9, ..., 2 decodiert.

Infobit 1 ist im Beispiel ein Sonderfall: Hier wird nicht über einen Vergleich verschiedener Zellen, sondern (mehrheitlich) durch Vergleich mit dem Initialwert 0 entschieden.

Bei vor Decodierung ungestörten Zellen bleibt am Ende eine ganz mit 0 belegte Matrix übrig. Bei ursprünglich gestörten Zellen entsteht dann bei korrekter Decodierung am Ende ein Fehlerbild. Dieses Fehlerbild kann zur Qualitätsbeurteilung der Decodierung oder des Signals herangezogen werden.

Abgesehen von Infobit 1 werden also bei diesem Beispiel alle Vergleiche direkt zwischen aktuellen Zellinhalten realisiert. Würde man beispielsweise alle Zellen gegeneinander invertieren, so würden trotzdem alle Bits 2..10 korrekt decodiert werden.

Soweit die Grundidee - wobei bisher die einfache binäre Ungleich/Gleich-Abfrage, realisiert durch XOR, verwendet wurde. Die Verallgemeinerung auf Soft-Decision-Mehrheitsentscheidungen geschieht durch Übergang von konventioneller Logik zu einer Fuzzy Logik, bei der XOR ersetzt wird durch $XOR(A,B) = ABS(A-B)+1$; ($ABS = \text{Absolut-Betrag}$). Zu Details siehe [7].

4 Lokalisierung

Für Codes mit freier Drehlage werden zur Beschleunigung an verschiedenen Stellen kleine spezielle Muster eingestreut, die zur Lokalisierung dienen. Durch das redundante Einstreuen von kleinen Lokalisierungsmustern wird das Lokalisieren robust gegen lokale Störungen, insbesondere auch gegen die in der Praxis häufig vorkommenden Störungen am Rand. Die Forderung einer Quiet Zone entfällt.

Von anderen Matrix-Codearten ist das Einstreuen von speziellen Mustern bekannt (ineinander verschachtelte Quadrate: Aztec, QR). Im Gegensatz dazu werden hier mehrere kleine Schachbrettmuster eingestreut. Das Finden dieser Muster ist ein hierarchischer, iterativer Vorgang, bei dem auf einen sehr einfachen ersten lokalen Operator ein zweiter, komplexerer folgt, mit strukturell mächtigeren Formen. Das Verfahren ist nicht auf eine Vorab-Binarisierung angewiesen, arbeitet differentiell und ist somit robust. Für Details sei auf das entsprechende Patent [6] hingewiesen. Die Muster bieten maximale Strukturiertheit auf kleinem Raum, dadurch wird möglichst wenig des kostbaren Codeplatzes für das Lokalisieren aufgebraucht. Die Muster unterscheiden sich optisch nicht von den informationstragenden Daten, so dass sich für das Auge eine unauffällige, insgesamt homogene optische Erscheinungsweise ergibt.

5 Anwendungsbeispiele

Die folgenden Anwendungsbeispiele seien herausgegriffen: Ein Hersteller von Dichtmitteln bietet für Zwecke der Produkt-Rückverfolgung und für Plagiatschutz individuell mit ECCi gekennzeichnete Radialwellendichtringe an [8]. Ein Lieferant für spezielle Laser-Bearbeitungseinrichtungen bietet die ECCi-Kennzeichnung und das Lesen von Bearbeitungswerkzeugen an, zur Vereinfachung der Handhabung der routinemäßigen Nachbearbeitung [9]. ECCi-Beschriftungen sind bei Laserbeschriftern bereits in die Beschriftungssoftware als Standard integriert [10].

Die Abbildungen 1.4 und 1.5 zeigen Beschriftungs- und Leseproben mit Radialwellen-Dichtringen und mit Bearbeitungswerkzeugen.

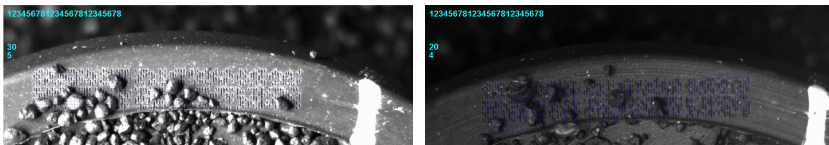


Abbildung 1.4: Individuelle Kennzeichnung von Simmerringen (Leseprobe mit verschiedenen Störungen und Beleuchtungen)

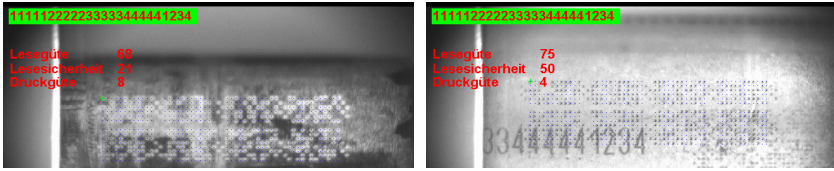


Abbildung 1.5: Individuelle Kennzeichnung von Bearbeitungswerkzeugen (Beschriftungs- und Leseprobe mit versch. Beschriftung und mit Störungen)

6 Vergleich mit Alternativen

Die Codierung ist differentiell; damit adaptiert die Decodierung automatisch an beleuchtungsbedingte Veränderungen der Bilddarstellung. Die Quiet Zone entfällt bei ECCi; das gesamte Muster wird zum Lokalisieren mit herangezogen, daher ist die Lokalisierung robuster und genauer als bei ECC 200. Die Fehlertoleranz ist bei ECCi deutlich höher als bei Data Matrix. Die Datendichte ist bezogen auf die Anzahl der Zellen bei ECCi geringer als bei Data Matrix (je nach Parametrierung ab Faktor 2).

Bei alternativen Methoden, die mit 90 Grad gegeneinander steigenden/fallenden Strichen („Glyphen“) arbeiten [11], geben diese dem Lesesystem zwar einfache Hinweise für die Drehlage, ihre Darstellung ist jedoch bei verformender Beschriftung empfindlich gegen die Beleuchtungsrichtung. Der Aufwand für die Beschriftung ist bei ECCi geringer, denn bei ECCi müssen nur ca. 50% der Zellen beschriftet werden, ausserdem sind keine Formelemente erforderlich: bei ECCi genügt ein einziger Punkt. Dieses wiederum führt zu flächenbezogen größerer Datendichte.

7 Fazit

Der neue Code vom Typ ECCi ist interessant in Industrienanwendungen mit problematischer Oberfläche, bei kleinen bzw. nicht rechteckigen Beschriftungsfeldern, auf gekrümmten Oberflächen (angepasstes Format), bei schwieriger Zugänglichkeit durch die Beleuchtung, bei notwendigerweise kleinflächiger (punktuellem) Beleuchtung aus Betrachtungsrichtung.

Mit dem Code werden viele Probleme, die mit der Sicherstellung der Lesbarkeit von Data Matrix Codes auftreten und durch Normen und Verifikatoren nur unzureichend gelöst werden können, umgangen: Auf-

grund der Codierungs- und Decodierungsweise sind Markierungen prinzipiell bei allen realen Beleuchtungsbedingungen lesbar, d.h. wenn er nur überhaupt beleuchtet wird und sich irgendwie in einem Bild darstellt und nicht extrem gestört ist. Damit kann die Sicherstellung der Lesbarkeit gewährleistet werden über die gesamte Logistik-Kette, in verschiedenen Einbau-Situationen und über die gesamte Lebensdauer des Produkts. Bei Industrieanwendungen sind die zu codierenden Datenmengen in aller Regel geringer als bei Anwendungen auf Papier, daher kann hier mit höherer Redundanz, also geringerer Datendichte gearbeitet werden.

Literatur

1. Association for Automatic Identification and Mobility (AIM) specification for Data Matrix. [Online]. Available: www.aimglobal.org/estore/ProductDetails.aspx?productID=26
2. "DIN ISO 15415; Europäische Norm-Data Matrix Quality Requirements for Parts Marking-EN 9132; AIM DPM-1-2006-Direct Part Mark Quality Guideline; Applying Data Matrix Identification Symbols on Aerospace Parts (standards.nasa.gov/documents/viewdoc/3314928/3314928)."
3. Understanding 2D Verification. [Online]. Available: reynolds-automation.com.au/Microscan/Literature/Training/2Dverification_whitepaper.pdf
4. "Direktbeschriftung mit Data Matrix Codes (DMC) - Qualitätsvorgaben," Firmenschrift, Vision Tools Bildanalyse Systeme GmbH.
5. W. Proß, F. Quint, and M. Otesteanu, "Using PEG-LDPC Codes for Object Identification," in *Int. Symp. on Electronics and Telecommunication (ISETC'2010)*, Timisoara, 1993.
6. "Patent DE 10220220 C1."
7. "Patente US 7411524 B2, DE 10 2005 037 388 B4."
8. "Freudenberg Save," Pressemeldung der Firma Freudenberg, Weinheim. [Online]. Available: www.freudenberg-ds.de/ecomaXL/index.php?site=FDS.de_news_detail&udtx_id=306
9. "Fa. Kist Lasertechnik," Eckental. [Online]. Available: www.kist-lasertechnik.de
10. "ROFIN and VisionTools to Showcase Innovative DataMatrix Code for Industrial Marking Applications," Pressemeldung der Firma Rofin Sinar, Bergkirchen. [Online]. Available: www.rofin.com/uploads/media/ECCLCodes_English.pdf
11. "Patent US 38758 E."