mehr zum thema:
www.scch.at/de/multitouch-user-interface

MASCHINEN WIE EINE APP BEDIENEN: ANFORDERUNGEN AN MULTITOUCH-BEDIENUNG IN DER INDUSTRIE

Nicht nur Smartphones und Tablets werden per Touchscreen bedient. Auch in Maschinensteuerungen und Industrieanlagen abseits von Consumer-Electronics greift man seit nahezu 20 Jahren auf berührungssensitive Bildschirme zurück. Allerdings kommen dort bisher relativ ungenaue, resistive Touchscreens zum Einsatz, die nicht in der Lage sind, mehrere gleichzeitige Berührungen zu verarbeiten. Durch den zunehmenden Einsatz der neuen kapazitiven Touchscreens auch in industriellen Benutzungsschnittstellen ergeben sich völlig neue Interaktionsmöglichkeiten, wie etwa Mehrfinger-Gesten, die man bereits von iPhone & Co kennt. In diesem Artikel wollen wir allgemeine Anforderungen und Herausforderungen im Einsatz von Multitouch für den industriellen Einsatz diskutieren.

Trotz zunehmender Autonomie von industriellen Maschinen und Anlagen spielt die Schnittstelle zwischen Mensch und Maschine eine entscheidende Rolle bei der Vermeidung von Fehlern und der Optimierung von Produktionsabläufen. Neben neuen Trends, wie Sprachsteuerung, *Augmented Interfaces*[1]) oder *Tangible User Interfaces*[2]), setzt die Industrie seit mehr als 20 Jahren hauptsächlich auf resistive Touchscreens, um grafische Benutzungsschnittstellen zu realisieren. Verglichen mit den heute in Smartphones üblichen kapazitiven Touchscreens, sind resistive Touchscreens weniger sensibel und können nur mit sehr aufwändigen Technologien mehrere gleichzeitige Berührungen verarbeiten. Diese Beschränkung führte in der Vergangenheit dazu, dass resistive Touchscreens meist nur als Mausersatz dienten. Die Bedienung einer grafischen Benutzungsschnittstelle mit mehreren Fingern sowie die Verwendung von komplexen Gesten waren mit resistiven Touchscreens nicht möglich.

Ausgelöst durch den Siegeszug von Apples iPhone und iPad im privaten Anwendungsbereich, fielen zuletzt die Preise für hochsensible kapazitive Touchscreens. Diese leiden nicht unter den Einschränkungen von resistiven Displays und ermöglichen somit eine echte Multitouch-Bedienung. Durch die weite Verbreitung von innovativen Multitouch-Displays wird die intuitive und attraktive Gestaltung von grafischen Benutzungsschnittstellen auch in industriellen Anwendungen immer stärker nachgefragt.

Bisher wurden bei eingebetteten Systemen – aufgrund ihrer limitierten Speicherkapazität und Rechenleistung – die Interaktionsmöglichkeiten oft auf das Minimum beschränkt. Viele dieser in der Industrie traditionellen, formularbasierten Benutzungsschnittstellen werden in den nächsten Jahren durch moderne Interaktionskonzepte sowie moderne UI-Frameworks abgelöst. Hauptmotivation für die Ablösung dieser häufig überladenen formularbasierten Maschinenschnittstellen ist der Wunsch der Bediener nach einer aufgeräumten grafischen Benutzungsschnittstelle. Da die Menge der Softwarefunktionen an der Hardware einer Maschine für einen Käufer nicht sichtbar ist, wurde die grafische Benutzungsschnittstelle mit einer maximalen Menge an Eingabemöglichkeiten und Funktionen überladen, um deren Existenz zu beweisen.

Hier scheint eindeutig der Zenit erreicht, da viele Benutzer heute intuitivere grafische Benutzungsschnittstellen fordern, die stärker auf wenige zentrale Aufgaben einer Maschine fokussiert sind. Gerade hier bietet die Bedienung mit Hilfe von Multitouch-Gesten enormes Potenzial für grafische Benutzungsschnittstellen, die intuitiver und flüssiger zu bedienen sind als herkömmliche Benutzungsoberflächen.

Die technische Möglichkeit, Multitouch-Gesten in industriellen grafischen Benutzungsschnittstellen zum Einsatz zu bringen, wirft jedoch noch einige grundlegende Fragen für zukünftige industrielle Interaktionsdesigns auf. Diese werden wir in diesem Artikel näher betrachten.

[1]) *Wissenschaftlicher Fachbegriff für Schnittstellen, die durch Projektion entstehen, z. B. Brillen, Projektoren.*

[2]) *Wissenschaftlicher Fachbegriff für greifbare und fühlbare Schnittstellen.*

die autoren

Wolfgang Beer
(Wolfgang.Beer@scch.at)
ist Leiter des Forschungsbereichs Software Engineering und Technologie am Software Competence Center Hagenberg. Sein besonderes Forschungsinteresse gilt mobilen und eingebetteten Systemen, kontextsensitiven Systemen sowie industrieller Software.

Heinz Stummer
(stu@keba.com)
beschäftigt sich bei der KEBA AG seit mehreren Jahren mit Innovationen für Benutzungsschnittstellen.

Mario Winterer
(mario.winterer@scch.at)
ist wissenschaftlicher Mitarbeiter und Projektleiter am Software Compentence Center Hagenberg. Er beschäftigt sich seit etwa 12 Jahren mit Rich-Client-Anwendungen, Client/Server-Systemen und industrieller Software.

Anforderungen an industrielle Benutzungsschnittstellen

Für den Einsatz in industriellen Anwendungen müssen grafische Benutzungsschnittstellen spezielle Anforderungen erfüllen. Diese resultieren aus den erforderlichen Sicherheitsstandards, langen Produktlebenszyklen sowie der rauen Umgebung, in der diese Schnittstellen zum Einsatz kommen. Einige dieser speziellen Anforderungen wollen wir hier im Detail diskutieren.

Langlebigkeit

Industrielle Maschinen und Anlagen haben im Vergleich zu anderen Produkten aufgrund der hohen Anschaffungskosten erheblich längere Produktlebenszyklen. Während man bei Smartphones von einer Lebensdauer unter zwei Jahren spricht, kommt man bei einer Produktionsmaschine auf eine weitaus höhere Lebensdauer von bis zu 25 Jahren. Um diese langen Lebenszyklen garantieren zu können, muss auch die Basissoftware entsprechend wartbar und flexibel gestaltet sein. Zugekaufte und von Herstellern nicht mehr unterstützte Software-Bibliotheken können hier schnell zu einem Problem werden.

Verfügbarkeit

Eine der wichtigsten Anforderungen in der industriellen Anwendung ist die Vermeidung von Stehzeiten. Der Ausfall einer Maschine in einer Produktionskette kann zum Stillstand der gesamten Anlage und damit zu einem beträchtlichen finanziellen Schaden führen.

Robust gegen Fehlbedienung

Bei der Steuerung von Industrie-Hardware, wie zum Beispiel Achsen und Antrieben, kann bei einer Fehlbedienung erheblicher Schaden an Mensch und Maschine entstehen. Daher enthalten viele grafische Benutzungsschnittstellen in der Industrie Mechanismen, die eine Fehlbedienung verhindern sollen. Viele Hersteller setzen hier auf taktiles Feedback, explizite Gestik oder traditionellen Zweihand-Betrieb.

Bedienbarkeit in der Produktion

Industrielle Benutzungsschnittstellen werden häufig unter äußerst widrigen Umgebungsbedingungen eingesetzt. Im 24/7-Produktionsbetrieb muss eine Touch-Oberfläche mit Handschuhen bedienbar und tolerant gegenüber grober Verschmutzung, Staub, extremen Temperaturschwankungen und Feuchtigkeit sein. In manchen Anwendungsbereichen sind auch EMV[3])-Normen und Explosionssicherheit der Elektronik ein Kriterium.

Blindbedienung

Bei der Bedienung von Werkzeugmaschinen braucht der Benutzer beim Ausführen von kritischen Bewegungen den direkten Blickkontakt zum Prozess, d. h. er muss die zur Ausführung dieser Bewegungen notwendigen Bedienelemente nach einer kurzen Orientierungsphase erspüren und bewusst auslösen können, ohne den Blickkontakt vom Prozess abzuwenden.

Performanz

Die Reaktionsgeschwindigkeit der Benutzungsschnittstelle sollte eine Aktion an einer Maschine möglichst zeitnahe wiedergeben. Ein Benutzer muss das Gefühl haben, dass eine Maschine sofort auf Kommandos reagiert.

Herausforderungen

Aufgrund der beschriebenen allgemeinen Anforderungen an industrielle Benutzungsschnittstellen ergeben sich einige Herausforderungen sowohl für die eingesetzte Hardware als auch für die Software.

Herausforderungen aus Sicht der Hardware

Kapazitive Touchscreens registrieren einen Touch-Punkt durch die Berührung eines leitenden Objekts, wie einem Finger. Dadurch ist – im Gegensatz zu resistiven Touchscreens – kein tatsächliches Drücken erforderlich, wodurch eine deutlich sensiblere und flüssigere Eingabe möglich wird, aber auch die Gefahr einer ungewollten Bedienung entsprechend erhöht wird und eine Blindbedienung ohne zusätzliche Maßnahmen unmöglich ist.

Nachteil der kapazitiven Technologie ist auch, dass der Benutzer keine herkömmlichen Eingabestifte als Eingabeunterstützung verwenden kann. Auch die Bedienung mit Handschuhen stellt eine Herausforderung dar, da Handschuhe die Leitfähigkeit der Finger reduzieren. Abhilfe schaffen hier speziell für kapazitive Touchscreens entwickelte Eingabestifte und Handschuhe sowie eine neue Generation von kapazitiven Touchscreens, die eine wesentlich höhere Sensibilität bieten. Je höher die Empfindlichkeit eines Touchscreens ist, desto störanfälliger ist der Bildschirm prinzipiell gegenüber feldgebundenen Störungen, die im industriellen Umfeld sehr häufig sind.

Eine weitere Herausforderung für den Einsatz im Produktionsumfeld stellen Wasserspritzer auf der Touch-Oberfläche dar. Ein Wassertropfen genügt, um das Ergebnis der Touch-Punkt-Auswertung vollständig zu verfälschen. Eine Lösung, um dieses Problem in den Griff zu bekommen, ist die automatische Erkennung und Filterung von untypischen Eingabemustern (vgl. [Mat12]).

Dank der kapazitiven Technologie ist es möglich, mit geringer Einschränkung der Bedienbarkeit zusätzliche Scheiben vor dem Touchscreen zu montieren. Das erhöht den physikalischen Schutz und ermöglicht das Anbringen von Beschriftungen, verstärkt jedoch den Parallaxen-Fehler, der entsteht, wenn der Benutzer den Bildschirm berührt und dabei nicht im rechten Winkel darauf blickt. Je nach Stärke der vormontierten Scheibe und des Betrachtungswinkels können sich erhebliche Differenzen zwischen tatsächlicher und beobachteter Position des Berührungspunktes ergeben.

Bei der Auswahl der Hardware sind aus unserer Sicht auch die Reaktionszeit des Touchscreens und die maximale Anzahl gleichzeitig erfassbarer Touch-Punkte wesentlich. Diese Kennzahlen bestimmen in großem Maße die Flüssigkeit der Interaktion und die Menge der erfassbaren Gesten. Ein weiteres wesentliches Qualitätsmerkmal bei Touchscreens ist der minimal notwendige Abstand, um zwei Touch-Punkte am Sensor zu erkennen. Der geforderte Abstand von weniger als 12 mm wird heute insbesondere von sehr großen Touchscreens oft noch nicht erfüllt.

Herausforderungen aus Sicht der Software

Bei der Entwicklung von unterschiedlichen Multitouch-Benutzungsschnittstellen für industrielle Anwendungen stieß unser Team auf zahlreiche Herausforderungen aus Sicht der Software sowie der Software-Frameworks. Viele dieser Aspekte ergeben sich aus der Tatsache, dass resistive Touchscreens bisher hauptsächlich als Ersatz für eine Eingabe-Maus dienten. Dadurch und auf Grund der Tatsache, dass bisher nur eine Sensorkoordinate zu einem

[3]) *Elektromagnetische Verträglichkeit von elektronischen Geräten und Bauteilen.*

Zeitpunkt unterstützt wurde, ist die Multitouch-Fähigkeit bisher nur in wenigen Frameworks berücksichtigt.

Eine weitere Herausforderung ist die Koexistenz von Touch- und Maus-Bedienung. In den meisten Fällen muss eine Anwendung sowohl durch Multitouch als auch mittels Maus und Tastatur bedienbar sein. So muss zum Beispiel für eine Zwei-Finger-Zoom-Geste auch eine äquivalente Möglichkeit zur Bedienung mit Maus und eventuell Tastatur vorgesehen werden. Da industrielle Benutzungsschnittstellen stark auf maskenorientierte Oberflächen setzen, stellt die Verwendung von Multitouch-Gesten zusätzliche Herausforderungen an den Interaktionsdesigner. Eine grafische Benutzungsschnittstelle besteht aus einer Menge von hierarchisch angeordneten Elementen (*Widgets*). Jedes dieser Elemente kann selbst lokaler Empfänger von Events (beispielsweise Touch- oder Maus-Events) sein. Da eine Multitouch-Geste sich oft über mehrere lokale Elemente erstreckt, muss der Designer entscheiden, ob ein Ereignis für ein lokales Element zugestellt werden soll oder ob es schon Teil einer globalen Multitouch-Geste ist. Nur ein durchgängiges Gesamtkonzept hilft hier, die Mehrdeutigkeit von lokalen Klicks zu Multitouch-Gesten aufzulösen.

Auch das darunter liegende Betriebssystem muss die Weitergabe der Multitouch-Events unterstützen. Viele Windows- und Linux-Varianten, die in industriellen Geräten eingesetzt werden, unterstützen Multitouch allerdings nicht.

Ein vom Entwickler ausgewähltes Framework muss zudem eine variable Anzahl von erfassbaren Touch-Punkten unterstützen. Die derzeit erhältliche Hardware erkennt bis zu 30 und mehr gleichzeitige Berührungspunkte. Die Interpretation dieser Menge an Touch-Punkten erfolgt dann durch ein Framework zur Gestenerkennung. Neben den heute bekannten Grundgesten, die meist bereits vom Betriebssystem identifiziert werden, sollte ein Software-Framework zudem auch Unterstützung für die Erkennung von benutzerdefinierten Gesten anbieten. In **Abbildung 1** sind drei Beispiele dieser Grundgesten dargestellt. Die Abbildung zeigt eine Zoom-Geste mit zwei Fingern, eine Geste zur Rotation mit Hilfe von zwei Händen sowie eine Drag-Geste, die mit einer ganzen Hand durchgeführt wird. Neben diesen drei Beispielen existiert noch eine ganze Reihe von Standardgesten, die für unterschiedlichste Aktionen eingesetzt werden.

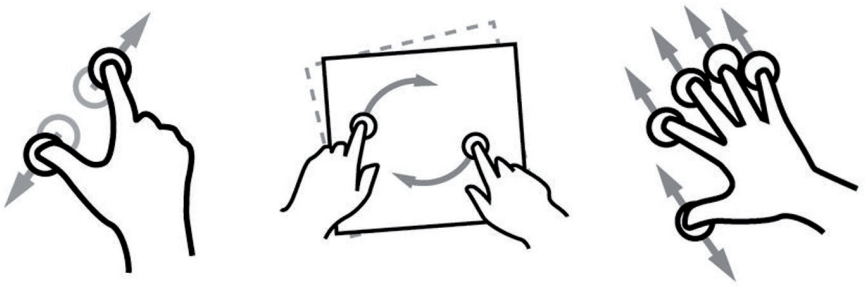

Abb. 1: *Zoom-, Rotate- und Drag-Gesten. (Bildquelle: GestureWorks)*

Bei der Entwicklung von grafischen Benutzungsschnittstellen für die Industrie sollte auf jeden Fall auch überprüft werden, ob das Software-Framework auch eine in der Zielhardware eventuell vorhandene 3D-Hardwarebeschleunigung optimal unterstützt. Die heute in der Consumer-Elektronik üblichen grafischen Effekte, wie etwa flüssiges Scrollen in Listen, aufwändige Schatten, Spiegelungen, semitransparente Flächen oder das animierte Rotieren, Kippen und Skalieren von grafischen Elementen, sind ohne 3D-Beschleunigung nicht ruckelfrei zu realisieren.

Aktuelle Technologien

Nach der Diskussion der Anforderungen an industrielle Benutzungsschnittstellen stellt sich die Frage, welches aktuelle Framework besonders für die Implementierung einer Multitouch-Anwendung geeignet ist. Wir wollen in diesem Abschnitt jedoch nur auf wenige, sehr populäre Frameworks (vgl. [Kam10]) eingehen.

JavaFX

Oracle bietet mit „JavaFX“ das neue zukünftige Java-UI-Framework für plattformübergreifende Rich-Client Applikationen. JavaFX ist seit der Version 2.2 im Java Development Kit 7 (ab Version 1.7_07) enthalten und für die Plattformen Windows (Vista, XP und Windows 7), Linux (Ubuntu 10.4+, GTK 2.18+) und Mac (OS X) verfügbar. Die Performanz ist stark von der Verfügbarkeit einer 3D-Hardwarebeschleunigung und entsprechender Treiber abhängig. Die aktuellen GPUs der gängigsten Grafikkarten-Hersteller werden allerdings bereits unterstützt. JavaFX bietet eine Programmierschnittstelle für Multitouch-Events und die Grundgesten „Swipe“, „Rotate“ und „Zoom“. Das Event-System kann flexibel um eigene Events erweitert werden, spezielle Strukturen für die Implementierung fortgeschrittener Gesten-Erkennungsalgorithmen gibt es jedoch nicht.

JavaFX ist besonders für Unternehmen interessant, die ihre Anwendung bereits in Java entwickeln und verschiedene Zielplattformen unterstützen wollen.

WPF

Microsofts „Windows Presentation Foundation“ (WPF) profitiert von der perfekten Integration in das Betriebssystem Windows. WPF unterstützt Entwickler mit einer Reihe von Standard-Multitouch-Gesten, die direkt in einer Anwendung verwendet werden können. WPF basiert auf Microsofts .NET Framework und kann somit mit jeder .NET-Sprache entwickelt werden. Als Nachteil erweist sich jedoch

Abb. 2: *Beispiel für ein aktuelles Bedien-Panel einer Spritzgieß-Maschine.*

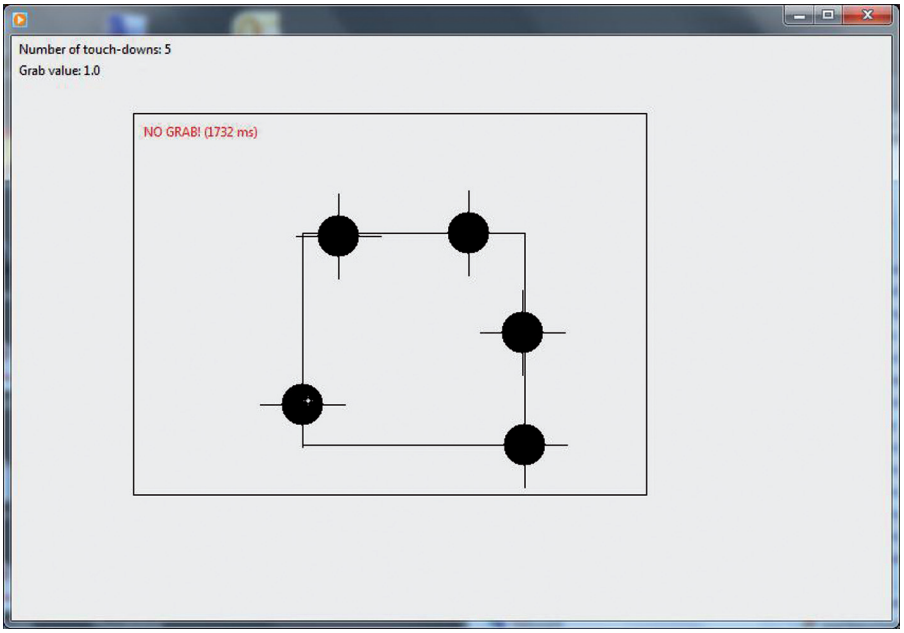

Abb. 4: Analyse einer möglichen Grab-Geste.

die Limitierung auf Microsofts Betriebssystem Windows.

QT

Das auf C++ basierende QT-Framework erweist sich als performante und sehr innovative Alternative zu JavaFX und WPF. QT unterstützt eine Vielzahl von verschiedenen Betriebssystemen und Hardwareplattformen und bietet mit der neuen, deklarativen Skriptsprache Quick eine bequeme Möglichkeit, um Benutzungsschnittstellen zu definieren, ohne C++ programmieren zu müssen. Als besonderer Bonus lassen sich QT-Anwendungen auch für Android Tablet-PCs übersetzen. Die Multitouch-Unterstützung ist allerdings leider noch mangelhaft.

HTML5

Der zukünftige Standard für interaktive Web-Seiten und -Anwendungen bietet die Möglichkeit, industrielle Anwendungen direkt in einem Web-Browser auszuführen. HTML5 bietet grundsätzlich die Möglichkeit, Multitouch-Events zu empfangen, überlässt die detaillierte Unterstützung jedoch den davon abgeleiteten HTML5-JavaScript-Frameworks, wie zum Beispiel „jQuery“ (vgl. [jQu]). Diese Frameworks unterstützen Multitouch zum derzeitigen Stand jedoch noch unzureichend. Für zukünftige Entwicklungen ist HTML5 aber sicher eine Überlegung wert.

Fallstudie: Bedienung einer Spritzgieß-Maschine

Um Erfahrungen über die Verwendbarkeit von Multitouch-Bedienung im industriellen Einsatz zu sammeln wurde mit der KEBA AG eine Multitouch-Benutzungsschnittstelle für eine Spritzgieß-Maschine[4]) entwickelt.

Derzeit werden für die Bedienung von Spritzgieß-Maschinen resistive Touchscreens mit mechanischen Tasten für die Steuerung der Maschinenbewegungen verwendet. Die oberste Maskenhierarchie wird direkt über Buttons, denen Maschineneinheiten zugeordnet sind, ausgewählt. Auf den Masken sind vielfältige Elemente angeordnet, die unter anderem folgende Aufgaben erledigen:

- Prozess-Istwerte anzeigen.
- Parameter anzeigen und einstellen.
- Verschiedene Liniengrafiken von Prozess-Istwerten darstellen.

Die Maskenhierarchie ist sehr stark verschachtelt und kann weit über 100 Masken enthalten. Auf einer Maschine müssen mehrere 1.000 Prozesswerte angezeigt und eingestellt werden. In Abbildung 2 ist die aktuelle Benutzungsoberfläche einer Spritzgieß-Maschine dargestellt.

Durch den zukünftigen Einsatz von Multitouch-Bedienung sollen Anwender die Spritzgieß-Maschine noch schneller, sicherer und intuitiver bedienen können. Beispielsweise kann man über einfache Multitouch-Gesten schnell von einer Eingabemaske zur nächsten wechseln oder durch die Verwendung von Zweihand-Gesten auf die lästigen Sicherheitsabfragen und Dialoge („Sind Sie sicher?“) verzichten. In Abbildung 3 ist der Prototyp der neuen, durch Multitouch-Gesten bedienbaren Benutzungsoberfläche dargestellt.

Die KEBA AG ist noch einen Schritt weitergegangen und hat die Steuerung der Maschinenbewegungen in den Touchscreen

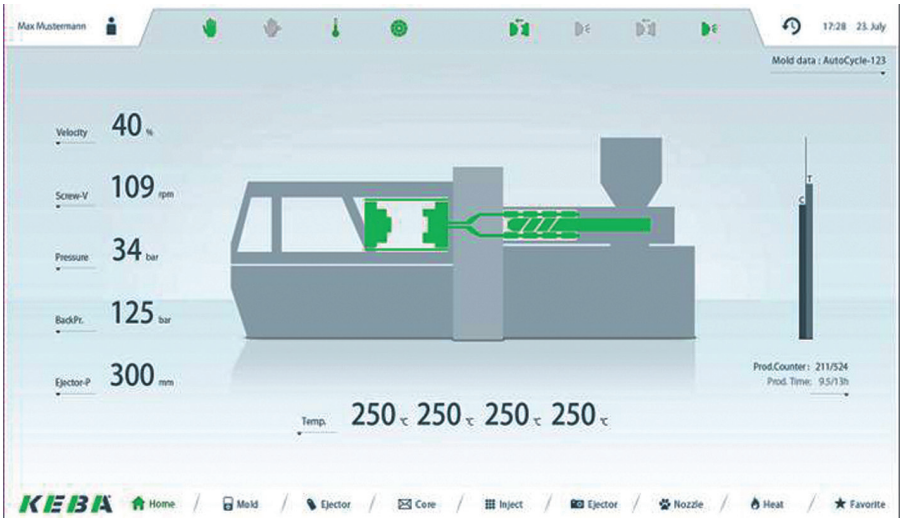

Abb. 3: Multitouch-Benutzungsschnittstelle einer Spritzgieß-Maschine.

[4]) *Eine Spritzgieß-Maschine wird verwendet, um Kunststoffgranulat unter hohem Druck zu verflüssigen und so in eine durch ein Werkzeug vorgegebene Form zu pressen. Die so erzeugten Kunststoff-Produkte reichen von filigranen Kontaktlinsen bis zu großen Regenwassertonnen.*

integriert. Durch die Kombination von Gesten und am Touchscreen aufgedruckten haptischen Elementen (ähnlich der Blindenschrift) können mechanische Tasten ersetzt werden, ohne die Bediensicherheit aufgeben zu müssen. Der Bediener hat dadurch ein haptisches Feedback, wenn er eine Maschinenachse verfährt, ob er die gewünschte Achse ausgewählt hat und ab wann er die Aktion tatsächlich auslöst. Auch eine Blindbedienung wird dadurch ermöglicht. Durch die Integration der Steuerung der Maschinenbewegungen in den Touchscreen sind eine intuitive Benutzerführung und eine vollkommen flexible Nutzung des Bedien-Panels möglich.

Besonderer Wert wurde beim Design auf eine intuitive und übersichtliche Benutzungsschnittstelle gelegt. Eine detaillierte Analyse der bereits bestehenden Eingabemöglichkeiten erlaubte es, Schlüsselaktionen für den Produktionsprozess zu erkennen und in ein völlig neues, konsistentes Interaktionskonzept zu integrieren. So wurde zum Beispiel darauf geachtet, Gesten in der Benutzerinteraktion über verschiedene Kontexte konsistent zu verwenden, um eine möglichst intuitive Bedienung durch den Maschinenbediener zu erreichen. Dieses Gesten-Framework wird in Kombination mit dem Visualisierungsframework in den kommenden KEBA Branchenlösungen eingesetzt und wird auch in Kombination mit der dazu passenden Hardware als HMI-Gesamtlösung erhältlich sein.

Aufgrund der speziellen Kombination von industrieller Hardware und Software musste auf bestehende Frameworks zur Gestenerkennung vollständig verzichtet werden. Somit wurden alle Multitouch-Gesten, die für das Interaktionskonzept notwendig waren, direkt implementiert. Dazu wurden die Bewegungsmuster der einzelnen Finger am Schirm über einen bestimmten Zeitraum analysiert, um auftretende Gesten zu identifizieren, wie in Abbildung 4 dargestellt ist. Wie gut zu erkennen ist, variieren mögliche Gesten in der Anzahl der beteiligten Finger, in den Bewegungsrichtungen der einzelnen Finger, sowie in der zeitlichen Abfolge der Bewegung. Bei der Definition der einzelnen Gesten wurde besonders darauf geachtet, ähnliche oder mehrdeutige Gesten zu vermeiden.

Fazit

Industrielle Benutzungsschnittstellen werden in den nächsten Jahren zunehmend auf die neue Multitouch-Technologie setzen und einige innovative Interaktionskonzepte hervorbringen. Diese Interaktionskonzepte werden intuitivere Benutzungsschnittstellen auch im industriellen Umfeld etablieren. Unter besonderer Berücksichtigung der Anforderungen an industrielle Benutzungsschnittstellen wird man mit Sicherheit in Zukunft auch Maschinen und Anlagen wie ein iPad bedienen können. ■

Literatur & Links

[jQu] jQuery, Multi-browser JavaScript library for client-side scripting of HTML5, siehe: jquery.com

[Kam10] D. Kammer, G. Freitag, M. Keck, M. Wacker, Taxonomy and Overview of Multi-touch Frameworks: Architecture, Scope and Features, in: Proc. of Workshop on Engineering Patterns for Multitouch Interfaces, Berlin 2010

[Mat12] J. Matero, A. Colley, Identifying unintentional touches on handheld touch screen devices, in: Proc. of the Designing Interactive Systems Conference, Juni 2012

[Yu10] N.-H. Yu, L.-W. Chan, L.-P. Cheng, M.Y. Chen, Y.-P. Hung, Enabling tangible interaction on capacitive touch panels, in: Proc. of the 23nd annual ACM symposium on User Interface Software and Technology (UIST 10), Oktober 2010