

## Embedded Vision – Neues Konzept mit neuen Anwendungen

Elektronische Komponenten werden immer kleiner und kostengünstiger. Das wird jedem sofort klar, der seinen Laptop/Tablet mit dem eigenen Rechner von vor 10 Jahren vergleicht oder die Größenentwicklung von Beamern, WLAN-Routern oder MP3-Playern vor Augen führt. Für Hersteller von Produkten oder Systemen wird es immer leichter und lukrativer, kleine Rechereinheiten für spezielle Teilaufgaben einzusetzen und diese im Gehäuse des übergeordneten Systems zu verbauen. Dadurch können Geräte nicht nur kleiner und günstiger werden, sondern auch mit neuen Funktionen ausgestattet werden. Eine solche Rechereinheit nennt man dann „eingebettet“ (engl.: embedded). Kommt zusätzlich auch Kameratechnik zum Einsatz, spricht man von „Embedded Vision“.

„Embedded Vision“ ist damit ein Teilbereich von „Computer Vision“ und bedeutet, dass ein System sehen und Bilder verarbeiten kann. Es ist kompakter und günstiger als ein klassisches Machine Vision System, also eines industriellen Bildverarbeitungssystems. Gleichzeitig erfüllt es aber dessen hohe Anforderungen an die Bildqualität und Spezifikationen.

In diesem White Paper möchten wir verschiedene technische Umsetzungen von Embedded Vision Systemen beleuchten.

### Inhalt

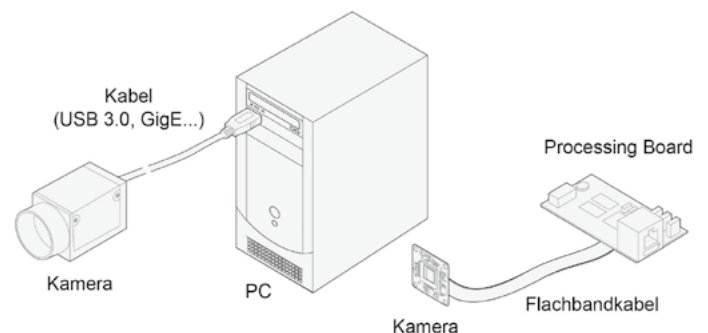
1. Embedded Vision Systeme.....	1
2. Eigenschaften von Embedded Vision Systemen .....	2
3. Die Hardwareseite bei Embedded Vision: Rechereinheiten.....	3
4. Möglichkeiten des Sehens: Die Kamera .....	3
5. Kostenersparnis dank Embedded Vision: die Softwareseite.....	4
6. Interfaces: Die Kameraanbindung .....	4
7. Welche Anwendungen von Embedded Systemen gibt es?.....	5
8. Fazit.....	6

### 1. Embedded Vision Systeme

Betrachten wir dazu zunächst ein klassisches Machine Vision System. In einem solchem System wird die Umwelt visuell aufgenommen und einer Rechenvorschrift folgend interpretiert. Der typische Aufbau einer solchen Lösung besteht aus einer Kamera mit Objektiv zur Bilderfassung, einem Kabel, das die Kamera mit einem PC verbindet, und dem PC selbst, oft einem Industrie-PC (IPC). Auf diesem findet die eigentliche Bildverarbeitung statt, die durch die entsprechende Bildverarbeitungssoftware gesteuert wird.

Die technologische Entwicklung der letzten Jahre macht es möglich, dass solche Bildverarbeitungslösungen, die bis dato nur mit hochklassigen Kamera- und PC-Plattformen realisiert werden konnten, nunmehr alternativ auch „embedded“ aufgebaut werden können. Als Rechereinheiten fungieren hier Single Board Computer (SBC), sogenannte System-on-Modules (SoM) oder aber auch individuell angefertigte Processing

Boards. Ein Embedded Vision System besteht dann z.B. aus einer Kamera ohne Gehäuse (einer sogenannten Boardlevel-Kamera), die über ein kurzes Kabel direkt an ein solches Processing Board angeschlossen wird. Dieses übernimmt die Aufgaben des PCs der klassischen Machine Vision Welt – aus Machine Vision wird Embedded Vision.



*Komponenten eines Machine Vision Systems im Vergleich zu den Komponenten eines Embedded Vision Systems*

Die Grenze zwischen Embedded Vision und Standard Machine Vision Systemen ist fließend und erscheint nicht immer ganz eindeutig. Denn ein PC, der in einer großen Maschine oder Anlage für genau eine Inspektionsaufgabe verbaut ist, ist im Prinzip auch eingebettet, wohingegen ein Single Board Computer (SBC) durchaus neben der Bildverarbeitung für weitere Ein- und Ausgabeverarbeitung sorgen und so das universelle Herzstück eines Gerätes darstellen kann.

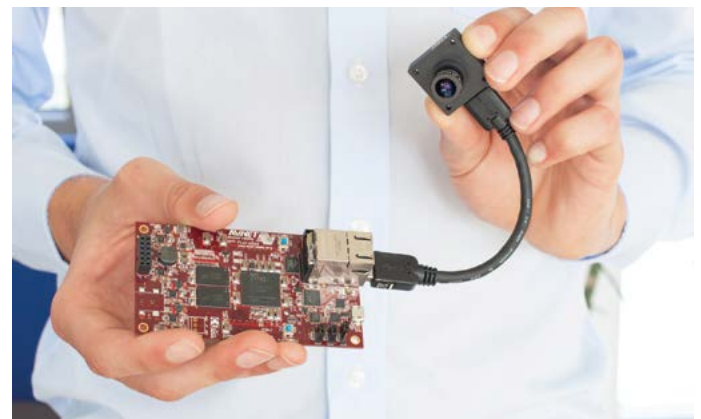
Es bietet sich (wie im Folgenden verwendet) an, wenn man das Embedded Vision System als Kombination aus einer kompakten (Boardlevel-) Kamera und einem Processing Board definiert, also auch ein SBC oder SoM.

## 2. Eigenschaften von Embedded Vision Systemen

Um die Eigenschaften von Embedded Vision Systemen genauer zu betrachten und im Vergleich mit einem klassischen Machine Vision System zu klassifizieren, sind hier vier Beispiele von Bildverarbeitungssystemen genannt:

	Beispiel 1: Standard Machine Vision System	Beispiel 2: Auf SBC basierendes Embedded Vision System mit Boardlevel-Kamera	Beispiel 3: Kompaktes Embedded Vision System	Beispiel 4: Hochspezialisiertes Embedded Vision System
Komponenten	IPC (1000 €) Kamera (500 €) Objektiv (250 €) Kabel (50 €) Betriebssystem (OS) (50 €) Software-Lizenz (150 €)	industrietauglicher SBC (280 €) USB-Kabel (20 €) USB-Boardlevel-Kamera (140 €) Objektiv (30 €) Betriebssystem (0 €) Software-Lizenz (150 €)	SoM mit Carrier Board (200 €) Kameramodul (100 €) Flachbandkabel (5 €) Objektiv (30 €) Linux-OS (0 €) Open Source Software (0€ )	individuelle Processing-Einheit (100 €) Kamera-modul (75 €) Flachbandkabel (5 €) Objektiv (20 €) Linux-OS (0 €) Open Source Software (0 € )
Stückkosten	Ca. 2500 €	Ca. 600 €	Ca. 335 €	Ca. 200 €
Anteil Standardkomponenten	Hoch	Mittel-Hoch	Teilweise	Gering
Entwicklungs- bzw. Integrationsaufwand	Gering: Stecker und Treiber bieten plug-and-play, direkte Anwendungsentwicklung	Gering-Mittel: Plug-and-play möglich, ggf. erhöhter Software-Entwicklungsaufwand	Mittel-Hoch: Kamera-Prozessor-Pipeline muss realisiert/ angepasst werden, spezielle Softwareanpassung, Hardwareentwicklung	Hoch: Kamera-Prozessor-Pipeline muss realisiert/ angepasst werden, spezielle Softwareanpassung, Hardwareentwicklung

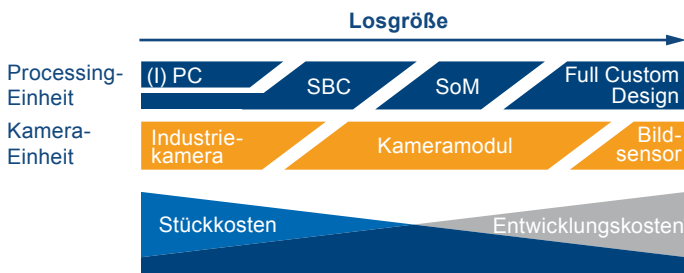
Das erste Beispiel stellt ein oben beschriebenes Standard Machine Vision System, basierend auf einem IPC, dar. Der Vorteil eines solchen Aufbaus liegt vor allem in seiner Einfachheit. Die auf Universalität und Vielseitigkeit ausgelegten standardisierten Bestandteile sind leicht zu beschaffen und lassen sich über Standardschnittstellen (z.B. USB, GigE) und Standardprotokolle (USB3 Vision, GigE Vision) per Plug-and-play miteinander verbinden. Ein Standardbetriebssystem, wie z.B. Windows, erlaubt eine zügige und effiziente Softwareentwicklung, ggf. mit Hilfe von kommerziellen Bildverarbeitungs-Softwarebibliotheken. Es ergeben sich Gesamtkosten von um die 2000 €.



Das zweite Beispiel beschreibt ein Vision System basierend auf einem SBC. Solche SBCs verfügen oftmals über USB-, mittlerweile auch USB 3.0-Anschlüsse. Es werden auch hier weitestgehend Standardkomponenten verwendet, so dass der Wert der Komponenten laut Stückliste bei deutlich geringeren ca. 600 € liegt - und das bei industrietauglicher Qualität. Bei solchen Lösungen wird meist ein Linux Betriebssystem verwendet. Vorteilhaft ist in diesem Zusammenhang, wenn der Kamerahersteller eine Plattform-unabhängige Kamerasoftware bereitstellt. Durch die bekannte Entwicklungsumgebung muss dann weniger neues Wissen aufgebaut werden und durch gleiche Schnittstellen wie auf dem PC-basiertem System kann sogar Applikationscode wieder benutzt werden.

Das dritte Beispiel ist ein Aufbau mit einem SoM mit Carrier Board (Trägerplatine) - und damit ein deutlich individuellerer bzw. stärker spezialisierter, aber dafür auch günstigerer Systemaufbau. Mit einer Boardlevel-Kamera, die auf einem MIPI CSI-2 oder LVDS-basiertem Interface (s.u.) basiert, können die Kosten auf ca. 335 € gesenkt werden. Die Integration ist aber aufwändiger, z.B. erfolgt die Softwareinstallation meist über eine Remote-Verbindung, somit müssen Applikationen am Desktop-Rechner entwickelt und nachfolgend auf das Zielsystem portiert werden. Auch ein erhöhter Aufwand für die Anpassung der Treiber bzw. Bildausleseroutinen kann erhöhte IT-Expertise und -Ressourcen erfordern. Dazu kommt die Entwicklung und Anpassung der Hardware, insbesondere das Design von einem Carrier Board.

Das vierte Beispiel zeigt den höchsten Entwicklungsbedarf, denn hier entstehen zusätzlich noch Aufwände für das komplette Design der Recheneinheit („Full Custom Design“). Damit werden aber auch – größere Stückzahl vorausgesetzt – die geringsten Hardware-Kosten erreicht, in unserer Beispielrechnung von 200 € – und das für ein System, das industrietaugliche Bildqualität liefert und entsprechend komplexe Bildverarbeitung durchführen kann.



*Kosten und geeignete Komponenten in Abhängigkeit der Losgröße*

Von Beispiel 1 bis 4 werden immer weniger standardisierte Komponenten verwendet. Entsprechend verringert sich der Anteil überflüssiger Hardwarekomponenten und der damit verbundenen Overhead-Kosten auf Seiten der Hardware. Jedoch wird gleichzeitig der Integrationsaufwand von 1 bis 4 höher. Dieser Aspekt wird im Folgenden genauer beleuchtet. Den noch spezielleren Fall einer Direktintegration eines Bildsensors betrachten wir hier nicht, verweisen aber auf das White Paper „*Ein Kameramodul für Embedded System: Make or Buy? Vor- und Nachteile einer eigenen Bildsensorintegration*“.

### 3. Die Hardwareseite bei Embedded Vision: Recheneinheiten

Als Recheneinheit in einem Embedded Vision System können im einfachsten Fall Single Board Computer eingesetzt werden, siehe Beispiel 2. Diese bieten durch ihre Bauweise als ein Modul einen Kostenvorteil gegenüber einem klassischen PC, sind aber im Umgang bei der Integration diesem relativ ähnlich. So werden oft gewohnte Schnittstellen und Betriebssysteme angeboten. Diese Einheiten sind als Consumerware sehr beliebt. Es gibt aber auch SBCs für professionelle, industrielle Anwendungen. Standardkomponenten erfordern weniger Hardwareanpassung als individualisierte Full Custom-Processing Boards. Letztere sind hardwareseitig vollkommen auf die Anwendung optimiert und daher am kosteneffektivsten. Sie sind aber nur mit hohem Aufwand und Fachwissen zu entwickeln. Einen sinnvollen Mittelweg für viele Applikationen stellen System-on-Modules (SoM) dar. Hierbei handelt es sich um Standardprodukte, die alle unverzichtbaren Rechnerkomponenten enthalten. Durch das Carrier Board, auf das die SoMs aufgesteckt werden, erfolgt die individuelle Anpassung an die Außenwelt, d. h. an Schnittstellen, Sensorik, Stromversorgung. In Summe

ergibt das dann das eigentliche Embedded Processing Board. Der Vorteil eines solchen Ansatzes liegt darin, dass der bei weitem komplexeste Teil der Hardwareentwicklung, nämlich das SoM, fertig zugekauft werden kann. Das Carrier Board, das im Wesentlichen das SoM nur noch an die externen Interfaces anbindet, fällt wesentlich weniger komplex aus und ist damit in der Entwicklung deutlich einfacher und kostengünstiger als ein Full Custom Design.

Hersteller von SoMs mit verschiedenen Prozessoren (bzw. System-on-Chips, SoCs), sowohl in x86- als auch ARM-Architektur, gibt es auch für industrielle Anwendungen in großer Zahl. Immer öfter werden auch Field Programmable Gate Arrays (FPGAs) verwendet – Prozessoren, deren Logikzellen frei programmiert und damit angepasst werden können. FPGAs stellen als Prozessoren für Vision Anwendungen oftmals die ideale Lösung dar, z.B. kann bei Stereovision- (3D) oder Recognition-Applikationen damit das effektivste Processing erreicht werden.

Die Hersteller designen ihre SoMs in der Regel anschlusskompatibel, so dass es ohne Neuentwicklung des Carrier Boards möglich ist, z.B. ein leistungsschwächeres SoM durch ein leistungsstärkeres zu ersetzen. SoMs können die Entwicklung eines Embedded Vision Systems auch schon bei geringeren Stückzahlen attraktiv machen. Auch wenn sich die günstigen Herstellkosten eines Full Custom Designs mit einem SoM-Ansatz wohl nicht erreichen lassen, gibt es gegenüber einem klassischen Standard PC-Setup immer noch einen gravierenden Kostenvorteil.



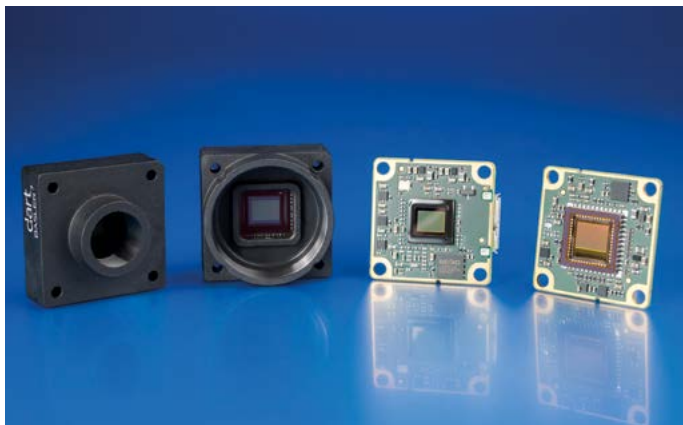
*Verschiedene Processing Boards*

### 4. Möglichkeiten des Sehens: die Kamera

Etlche Kameramodule, die auf dem Markt angeboten werden, stammen aus dem Handy, der Unterhaltungselektronik- oder dem Automotive Markt. Diese Module haben in der Regel eine unzureichende Bildqualität. Dies liegt daran, dass sie oftmals nur ein ‚schönes‘ Bild liefern müssen, das teilweise auch in der Applikation durch Softwareinsatz nachgeschönt und für das menschliche Auge optimiert wird. Für Computer Vision Anwendungen reicht diese Bildqualität hingegen oft nicht aus. Hier zählt der tatsächliche Informationsgehalt eines Bildes und hier stoßen diese Module teilweise an ihre Grenzen. Darüber hinaus stehen diese Kameramodule manchmal nur in großen Abnahmemengen zur Verfügung.

Viele Industriekamerahersteller bieten aber auch Kameramodule nach professionellem Standard und z. T. mit industriellen Sensoren für Embedded Anwendungen an. Solche Module bieten nicht nur eine Bildqualität, die für Machine Vision Anwendungen mit der entsprechenden Bildverarbeitung geeignet ist, sondern auch die Langzeitverfügbarkeit, die für die meisten Produkte notwendig ist. Solche professionellen Kameramodule werden oftmals auf einer einzelnen Platine mit oder ohne Objektivanschluss angeboten (Boardlevel-Kamera).

Ein weiteres Plus beim professionellen Kameramodul sind umfangreiche Bedienungs- bzw. Steuerungsmöglichkeiten, welche bei der Verwendung in einem System unabdingbar sind. Dazu zählen spezielle Trigger-Optionen (Software-/ Hardware-Trigger) oder Bildausgabeformate und integrierte Funktionen zur Bildvorverarbeitung (z.B. Debayering, Denoising, Fixed-Pattern Eliminierung etc.), die das Host-System bei der Bildverarbeitung deutlich entlasten. Dagegen liefern einfache Kameramodule nur die rohen Sensordaten, die danach noch optimiert oder verrechnet werden müssen.



Typische Boardlevel-Kameras für Embedded Vision Anwendungen

## 5. Kostenersparnis dank Embedded Vision: die Softwareseite



Einen Beitrag zur Kostenreduktion bei Embedded Systemen können frei verfügbare (z.B. als Open Source veröffentlichte) Softwarekomponenten bringen, neben Linux als Betriebssystem

z.B. OpenCV als Bildverarbeitungsbibliothek. Dadurch entfallen Lizenzkosten, die den Stückpreis erhöhen würden. Aber auch kommerzielle Software kann geeignet sein, die im Embedded Bereich teilweise mit eingeschränktem Funktionsumfang, aber auch günstiger angeboten wird.

Die Softwareentwicklung sehen viele Anwender berechtigterweise als einen der großen Posten im Entwicklungs-

budget solcher Systeme, denn diese ist für Embedded Systeme im Allgemeinen deutlich komplexer, aufwändiger und damit teurer als für klassische PCs. Die gute Nachricht: Viele Kamerahersteller liefern zu ihren Kameras eine Software-Schnittstelle (ein sogenanntes SDK = Software Development Kit), die es erlaubt, die Kamera in die eigene Software einzubinden und teilweise Sammlungen fertiger Programmteile zum Erstellen eines neuen Anwendungsprogramms enthalten. Diese helfen dabei, auch für Embedded Vision zu schnellen und robusten Ergebnissen zu kommen.

Viele Kamera-SDKs unterstützen sowohl Windows- als auch Linux-Betriebssysteme. Mittlerweile unterstützen viele dieser SDKs als Zielsysteme neben den klassischen, aus dem Standard PC-Bereich bekannten, x86-Prozessoren auch die weit verbreiteten ARM-basierten Architekturen. SDKs, die sowohl ARM- als auch x86-basierten Architekturen unterstützen, bieten dann einen deutlichen Kostenvorteil, wenn eine Portierung des Programmcodes von z.B. Windows/x86 nach Linux/ARM ohne großen Aufwand möglich ist. Denn die Wiederverwendbarkeit von bereits entwickeltem Code kann eine erhebliche Kostenersparnis bedeuten.

Hilfestellung oder sogar Entwicklungsvorlagen finden Entwickler oftmals auf offenen Webplattformen wie beispielsweise [github.com](https://github.com) oder [Imaginghub.com](https://imaginghub.com).

## 6. Interfaces: die Kameraanbindung

In klassische Machine Vision Systeme werden Kameras in der Regel über USB 3.0 oder GigE eingebunden. Der Vorteil dieser Interfaces sind ihre standardisierten Protokolle zur Kamerakonfiguration und Bilddatenübertragung, nämlich USB3 Vision bzw. GigE Vision. Nach der Installation eines (generischen) Treibers ist bei diesen Interfaces die Übertragung der Bilddaten soft- und hardwareseitig gesichert. USB 3.0 und GigE sind auch für den Embedded Vision Bereich interessant, insbesondere USB3 Vision. Sie stellen eine einfach zu integrierende Plug-and-Play-Lösung dar. Allerdings ist Ihre Bauform nicht besonders klein und ihre Datenübertragung kann den Prozessor zusätzlich belasten; Nachteile, die sich auch wieder in erhöhten Kosten niederschlagen können.



Typische Anbindung des Kameramoduls: USB- oder Flachbandkabel

Für eine stärker integrierte Anbindung bieten sich Board-zu-Board-Anschlüsse an, die durch ein Flachbandkabel das Kameramodul direkt mit dem Prozessor verbinden. Dadurch wird Overhead für Hardware und Datentransfer vermieden. Leider ist in diesem Fall aber wenig standardisiert.

So definiert z. B. der MIPI CSI-2 Standard der MIPI Alliance im Wesentlichen den physikalischen Layer und das Protokoll für den Bilddatentransfer. Hostseitige Treiberanbindung und API sind aber ebenso wenig Bestandteil der Spezifikation wie eine Hardwarestandardisierung (also ein einheitlicher Stecker). Dadurch ist eine jeweilige Komponentenanpassung auf dem (Carrier) Board unumgänglich. In der Regel bieten SoC-Hersteller Treiber für die Datenübertragung. Treiber für die Steuerung der Kamera sind aber immer SoC- und kamera- bzw. bildsensoren-spezifisch, d.h. generische Treiber sind nicht verfügbar und müssen systemspezifisch entwickelt werden.

Kamerahersteller können hier Vorteile für die Integration bieten, wenn Steuerungstreiber für einen SoC oder ein SoM zur Verfügung stehen. Ebenso ist es vorteilhaft, wenn sie für all ihre Modelle einen vereinheitlichten Output anbieten. Dann erfordert ein Kamerawechsel keine Neuintegration. Da der MIPI CSI-2 Standard dem mobilen Konsumgütermarkt entstammt, unterstützt er nicht alle Merkmale und Features von Industriekameras. Allerdings bieten hier verschiedene Hersteller wiederum technische Lösungen an, um diese Beeinträchtigungen auszugleichen. Eine starke Einschränkung von MIPI CSI-2 hinsichtlich des Produktdesigns bleiben jedoch maximale Kabellängen von nur ca. 20 cm.

Dagegen können mit Kameras, die LVDS- (Low Voltage Differential Signaling) basiert kommunizieren, Kabellängen von einigen Metern erreicht werden. Der LVDS-Standard der Datenübertragung bietet sich vor allem bei der Verwendung von FPGAs an. Auch für LVDS sind weder Stecker noch Datenübertragungsprotokoll standardisiert. Eine gute Integrierbarkeit einer LVDS-Kamera ist nur dann gegeben, wenn das Übertragungsprotokoll offen dokumentiert ist. Dann lässt sich die (systemspezifische) Ausleseroutine auf jedem FPGA-basiertem System integrieren und anpassen. Durch die Bereitstellung von IP-Cores (Logikmodule) für FPGAs sowie Referenzdesigns, z.B. in Form eines Development Kits kann die Kameraintegration unterstützt und erleichtert werden.

Softwareseitig kann die Anpassung auch dadurch vereinfacht werden, dass ein SDK mit festen Programmierschnittstellen (API) für den Anwender zur Verfügung steht, wie oben beschrieben.

Einige Hersteller bieten Komplettsysteme mit einem Kameramodul an, für das es bereits ein Processing Board mit einer (Software-) Integration über MIPI oder LVDS gibt. Allerdings ist der Benutzer dann sehr eingeschränkt bei der Auswahl der Komponenten, was nicht nur eine

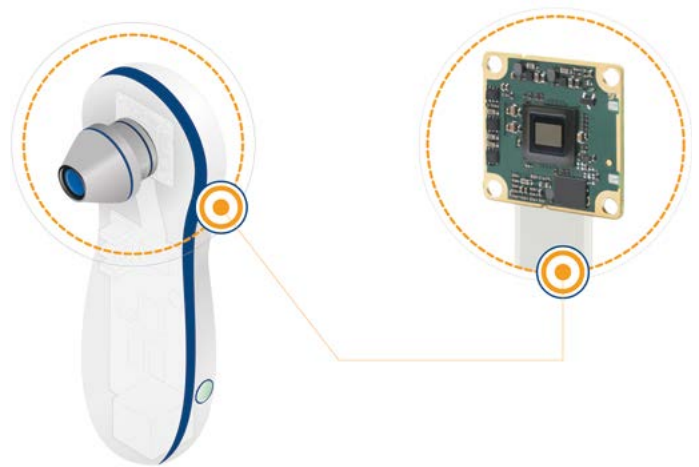
gewisse Abhängigkeit schafft, sondern es auch erschwert, die wirklich optimale Hardware einsetzen zu können.

## 7. Welche Anwendungen von Embedded Systemen gibt es?

Natürlich gibt es offensichtliche Vorteile eines Embedded Vision Systems wie den geringeren Platzbedarf und der relativ geringen Leistungsaufnahme, z. B. für mobile Systeme. Den wesentlichen Unterschied zu Machine Vision Systemen kann man hingegen so zusammenfassen: Embedded Vision Systeme haben deutlich geringere Stückkosten, erfordern aber in der Regel einen höheren Integrationsaufwand und damit größere Entwicklungskosten. Mittlerweile sind aber sowohl Hersteller von Processing Boards als auch Hersteller von Kameramodulen dabei, diese Integrationsaufwände zu verkleinern. Zum Beispiel indem Standards definiert oder universelle Integrationslösungen angeboten werden.

Dadurch, und weil die genannten Komponenten immer preisgünstiger werden, ist es einerseits möglich, günstigere Lösungen für bestehende Produkte bzw. Anwendungen zu schaffen, andererseits ergeben sich durch geringe Baugröße und geringen Preis auch viele neue Anwendungsmöglichkeiten, die erst durch Embedded Vision profitabel umgesetzt werden können.

Embedded Vision Systeme ermöglichen es somit, neue Märkte und Anwendungen zu erreichen, die ansonsten ohne Vision auskommen müssen. Neue kompakte und mobile Geräte der Medizindiagnostik, intelligente Systeme im Bereich Verkehr (Gefahrenerkennung), kleine Produkte für Smart-Home-Anwendungen oder smarte/mobile Module zur Fertigungsinspektion (Qualitätskontrolle) sind hierfür Beispiele.



*Embedded Vision Systeme lassen sich in vielen Produkten verwenden*

## 8. Fazit

Die Entwicklung von Embedded Vision Systemen wird bestimmt durch die Optimierung von Integrations- und Herstellungskosten.

Hardwareseitig punktet das Embedded Vision System hierbei dadurch, dass kostengünstige Standard Processing Boards, Modulkonzepte (SoM) oder gar Full Custom Boards verwendet werden können. Das verringert die Stückkosten – allerdings mit dem Nachteil höherer Entwicklungs- und Integrationskosten.

Für die Softwareentwicklung und Kameraanbindung müssen passende Treiber für die Kamerainterfaces sowie eine Programmierschnittstelle (SDK) zur Verfügung stehen, sowohl für Plug-and-play-Interfaces wie USB 3.0 als auch für stärker integrierte Anbindungen wie MIPI CSI-2 oder LVDS-basierte Schnittstellen. Auf dem Markt werden von einigen Kameraherstellern SDK-Pakete für x86 (Linux oder Windows) und ARM (Linux) angeboten, die mit optimierten Treibern und einer einheitlichen Programmierschnittstelle ausgestattet sind.

In den nächsten Jahren werden mehr und mehr Embedded Vision Lösungen auf dem Markt erscheinen. Sie sorgen für kleinere, leichtere und höher integrierte Designs, die immer öfter ohne klassischen PC auskommen. Sie ermöglichen auch, stationäre Vision Anwendungen mobil oder tragbar zu machen. Embedded Vision wird helfen, bestehende Vision Systeme preisgünstiger umzusetzen und zusätzlich neue Anwendungsfelder zu erschließen, die heute noch ganz ohne Kamertechnik auskommen müssen.



## Autor

**Dr. Thomas Rademacher**  
Product Market Manager  
- Factory & Traffic

Dr. Thomas Rademacher ist seit 2015 als Produktmanager bei Basler tätig. In dieser Funktion ist er verantwortlich für die Basler dart Kamera Serie und für Embedded Vision Systeme.

Nach Abschluss seiner Promotion im Fach Physik an der Universität Göttingen arbeitete er zunächst im Produktmanagement in einem führenden Unternehmen für industrielle Messtechnik in der Halbleiterbranche. Dort lag sein Tätigkeitsschwerpunkt im Bereich der optischen Metrologie und automatisierter Bildverarbeitung- und analyse.

## Kontakt

Dr. Thomas Rademacher  
Product Market Manager - Factory & Traffic

Tel. +49 4102 463 487  
Fax +49 4102 463 46487  
E-Mail: [thomas.rademacher@baslerweb.com](mailto:thomas.rademacher@baslerweb.com)

Basler AG  
An der Strusbek 60-62  
22926 Ahrensburg  
Deutschland

## Über Basler

Basler ist ein international führender Hersteller von hochwertigen digitalen Kameras und Zubehör für Anwendungen in Fabrikautomation, Medizin, Verkehr und einer Vielzahl von weiteren Märkten. Das Produktportfolio umfasst Flächen- und Zeilenkameras in kompakten Gehäusegrößen, Kameramodule als Boardlevel-Varianten für Embedded Vision-Lösungen sowie 3D-Kameras. Abgerundet wird das Angebot durch unser bedienerfreundliches pylon SDK sowie eine breites Spektrum von teils eigens entwickeltem Zubehör, das optimal auf unsere Kameras abgestimmt ist. Basler verfügt über drei Jahrzehnte Erfahrung im Bereich der Computer Vision. Das Unternehmen beschäftigt rund 500 Mitarbeiter an seinem Hauptsitz in Ahrensburg sowie in Niederlassungen und Vertriebsbüros in Europa, Asien und Nordamerika.