Smart Materials Detection Using Computer Vision

Conference Paper · September 2019

2 authors:

Mustafa Bilgin
University of Duisburg-Essen

Johannes Backhaus
Bergische Universität Wuppertal

Smart Materials Detection Using Computer Vision

Mustafa Bilgin, Johannes Backhaus

University of Wuppertal
Rainer-Gruenter-Straße 21
42119 Wuppertal, Germany

E-mail: bilgin@uni-wuppertal.de; jbackhaus@uni-wuppertal.de

Abstract
Smart materials detect autonomously changings of environmental conditions such as temperature (thermochromism), water / humidity (hydrochromism), UV/Vis light (photochromism), mechanical stress (piezochromism), acids (pH-chromic) and more. The irreversible colour changing of smart materials can be used to "translate" the particular colour information into the degree of contamination, such as a printable sensor. Integrated into a smart code, besides static data (2D) also dynamic sensor data (3D) can be stored. The aim of this study is the computer vision based image detection and interpretation of a smart material surface. The particular colour information will be transformed into the degree of contamination. The limits of feasibility will be examined on a laboratory scale and comparison with reference colour values.

Keywords: smart materials, neural networks, computer vision, smart code, piezoelectric inkjet

1. Introduction and Background

Smart Materials can show a hydrochromic (water/humidity), photochromic (UV/Vis light), thermochromic (temperature), piezochromic (pressure) and other behaviours. They gradually can shift between two states by changing their colour. Photochromic compounds show a light detecting behaviour. If the surface of a photochromic compound is exposed due to UV-light, it changes its colour gradually from one colour state to another colour state. The pH-chromic compound is used to determine the acid or base capacity of a substance. This e.g. can be used to monitor the freshness of foodstuffs. These materials are autonomous and independent of electrical circuits and electrical sources. A camera system can optically read the individual smart materials surfaces. Thus, by the recorded colour values can be concluded with a change in colour in terms of the degree of contamination. This colour changing behaviour can be used to develop printable sensors (Bilgin & Backhaus, 2017, a). That way, it is possible to store dynamic information (Smart Materials) about possible deviations and static consumer or product information inside a smart code. By means of computer vision, it is possible to automate tasks by processing, analysing, interpreting, and manipulating digital image information. This paper shows functions for recording a smart code and its subareas.

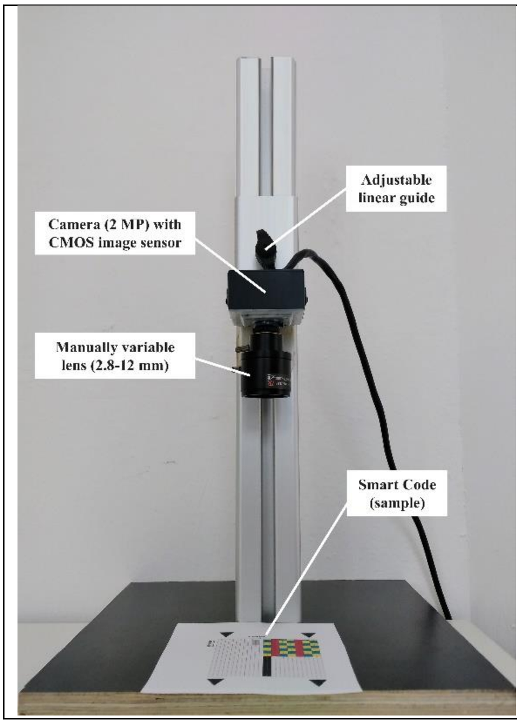
For evaluating a smart code, the following steps are necessary: Most of the code readers operate with a binarisation initiated by a prior grayscale transformation. This reduces the data volume with regard to short processing time. - It is also helpful to crop the image to a smaller size without the data getting lost - However, a grayscale transformation is not adequate for coloured sensor data as meaningful data can be lost. In this way, image files created in RGB can be processed in different colour conversions such as L*a*b*, which can be used as a reference colour system. It is important to use these colour systems to evaluate or calibrate the colours. Being JPG files, camera images often do not contain the device's profile (ICC), but either one of the standard profiles sRGB or Adobe RGB. These determinants must be controlled. The L*a*b* colour space contains all colours independent of any device. It therefore allows lossless conversion of colour information from one colour system to another, from one device type to another. This provides a subsequent interpretation of the colour information in statements about the degree of contamination. However, in order to correct the image information, the use of morphological operators can be useful. A further aspect is erosion, which causes false classifications to disappear due to erosion. These can be connected image areas. On the other hand, dilatation can be used to compress or strengthen partial areas in a pixel group, to fill holes, to close cracks. The combination of smart materials, smart devices and smart codes will allow the construction of an Internet of Things (IoT) in which all components communicate in an autonomous network to perform predetermined tasks, provide consumer information, collect feedback on their products and provide information on critical environmental impacts throughout the transport process (Ashton, 2009).

Objective of code development:
The Smart Code in this work is to be divided into three main areas: The static area for text content, the finder pattern area for identifying and aligning the code and the dynamic area for smart materials (printable sensors). The special feature of the static area is that the Braille font is to be used. With the help of the smartphone camera, visually impaired people or the unimpaired can read out the content of the code or touch it manually. In addition, the dynamic range in the smart code can be optically read out and results on the respective state of the smart materials can be output e.g. auditory.

2. Materials and Methods

2.1 Instruments

The smart code samples (containing smart materials) were printed with a piezoelectric inkjet printer (Epson WorkForce WF-3620). Technical parameters: Print Head: PrecisionCore; Thin Film Piezo element: 1/1000mm; Droplet Size: 2.8 pl (range of 1.5 – 32.5 picoliters); Nozzle Configuration: 800 Nozzles Black (K), 256 Nozzles per Colour (CMY); Printing Resolution: 4,800 x 2,400 DPI.



Experimental assembly
Technical parameters: The smart codes were captured with a 2.07 MP camera (ELP-USBFHD01M-BFV-D) with CMOS image sensor and a manually variable 2.8-12 mm lens - the focus of the camera is manually adjustable. The camera has an image resolution of 1080P (1920 × 1080 pixels).
The experimental setup is based on an adjustable linear guide. Thus, the camera can be moved vertically. In this publication, however, the lighting conditions are uncontrolled. However, the construction can be placed in a light chamber in order to realize the evaluation under a reference daylight Illuminant (e.g. D50 standard light). The recording of the samples is done in real time and under real conditions.

For reference purposes, samples were measured with a spectral densitometer (TECHKON SpectroDens) to analyse their characteristic RGB and CIEL*a*b values and to compare them with values detected and analysed by the computer vision. Technical parameters: polarising filter: off; type of light: D50, 2° standard observer; diameter of measuring orifice: 3 mm. This paper uses the OpenCV (Open Source Computer Vision Library), an open source computer vision and machine-learning library, for image processing. The algorithms can be used to recognize faces, identify objects, classify human actions in videos, track camera movements, track-moving objects, recognize scenes or, as in this case, detect smart codes and evaluate partial areas. The programs for analysing the smart codes were programmed in Python. For this purpose, other libraries like NumPy were used. NumPy extends Python with functions for scientific computing and numerical computation (OpenCV, 2019).

2.2 Standardization

All experiments were carried out under controlled laboratory conditions - reproducibility was ensured by an air conditioning system and deviations were recorded in protocols. Temperature: 20 °C (+/- 1 ° C); relative humidity: 55% (+/- 1%) were controlled.

2.3 Materials

All materials, which were used for this research, are listed in Table 1. To ensure the reproducibility, all experiments are based on standardised substrates Inapa tecno. The smart material used for the experiments is photosensitive Prussian blue, described in (Bilgin & Backhaus, 2018).

Substrates for printing	Inapa tecno, oxygen pure high-white recycled paper, Format: 210 x 297 mm (A4), Grammage: 80 g/m²
Dye	Photosensitive Prussian blue, CAS Number: 14038-43-8, Chemical formula: $C_{18}Fe_7N_{18}$, Molar mass: 859.24 g·mol^{-1}
Water-based base ink	E24, Octopus Fluids GmbH & Co KG, Colour: colourless, pH: 7,86, Conductivity: (mS/cm): <5, Viscosity (mPa·s): 3,00

Table 1 *Materials*

2.4 Test Chart

The initial intention was to modify a standard QR code and design dynamic areas (for smart materials) within the code. However, this idea was dropped because the QR codes are designed only for static data. Dynamic (i.e. information changing) areas avoid decoding the QR code. Consequently, a simple smart code containing static and dynamic areas was designed and prototyped in laboratory scale (Fig.1) The code is divided into three sections. The first area (red framed) shows three finder patterns (squares) in three corners. They are required to identify the smart code and determine the inner coordinate system of the code. The second area (grey hatched) is the static data area in which 54 characters of text information can be placed. The third area (yellow framed) is the dynamic area that consists of several cyan, magenta and yellow caches which here serve as placeholders for three different smart materials.

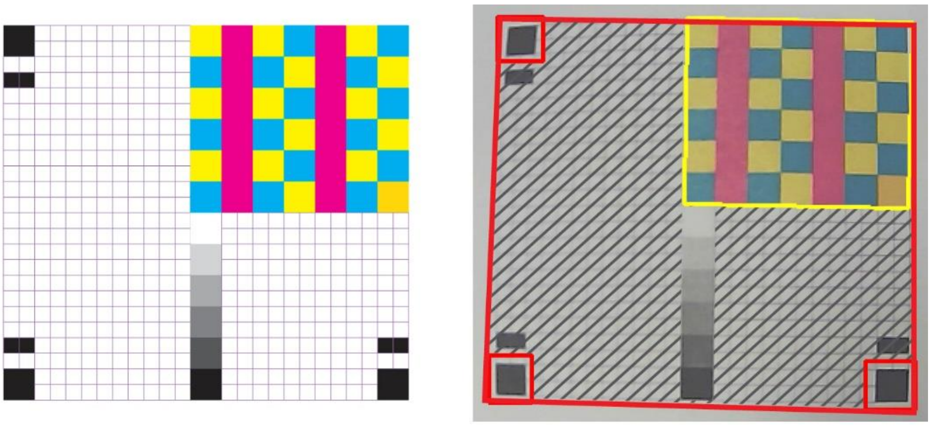


Fig. 1 The test chart shows a smart code prototype.

3. Results and Discussion

3.1 Static Data: Encoding and Decoding

There are many methods to encode information. Therefore, there are many coding schemes (Morse characters, nautical signal flags, semaphore, binary, numbers to letters conversion and more). However, this paper focuses on Braille (Fig.2). It usually consists of six embossed dots, but it was later adapted to computer language and consists of 8-point characters (International Standard ISO/IEC 10646 - Unicode) used by visually impaired people. This scheme allows 256 different characters to be displayed. A special requirement in the coding of the static content is that visually impaired people can read the content manually by touching it. However, the static data should also be readable and converted into an alphanumeric text (Latin alphabet) for persons who do not master the Braille writing process. - This can be done with the smartphone camera. The dynamic part of the code is also machine-readable and is evaluated by computer vision.

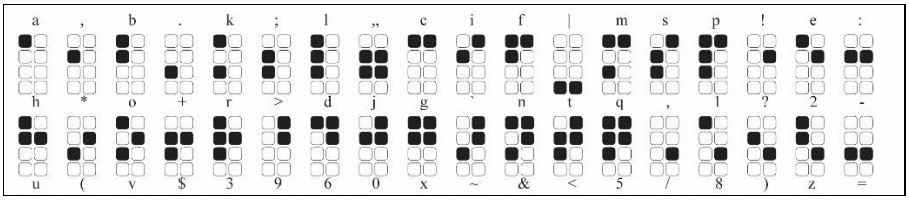


Fig. 2 Unicode 8-dot Braille characters

A usual scanner recognizes an image pixel by pixel. Here, a function is used that helps to identify points, whose dimensions, distances and respective colour information are required. The Hough Transformation (Hough and Paul, 1962) is a method for detecting straight lines, circles (such as the Braille points) or any other parameterizable geometric shapes in a binary black-and-white image. In the following example (Fig. 3) the Hough Circle method was applied. The result for the recognition of a Braille line and the application of the Hugh method must be prepared: Initially, an 8-bit image is converted to grayscalein order to reduce the entire image information to essential information. For the recognition of Braille points we refer to the OpenCV function cv2.HoughCircles in corresponds to Yuen et al. (1990).

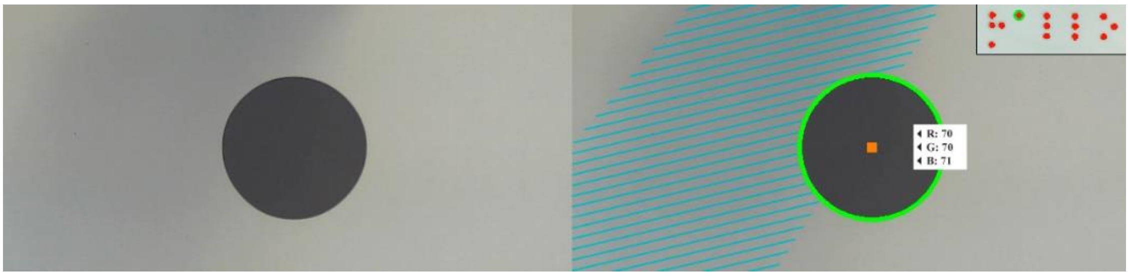


Fig. 3 Circle Hough Transform

The static area of the code is very important for a calibration of the camera in order to avoid external optical influences when analysing the colour of the sensitive dynamic code. By detecting the black points from the Braille scheme and the colour information of the white background, it is possible to use these information for a calibration of the camera. Each point of the Braille scheme and its adjacent background can be used to calibrate both the exposure rate and the white balance of the camera properties. The blue hatched area shows the influence of shadows, which can falsify the colour value. Influencing factors of ambient light and shade must be taken into account in a later real-time correction. The current actual value can be compared with the known target value of the individual points in order to identify deviations and use them for further statistical evaluations. The Braille point described in the example illustrates one of many possible colour information from varying Braille points used in the calibration area (chapter 3.4) to adjust the camera.

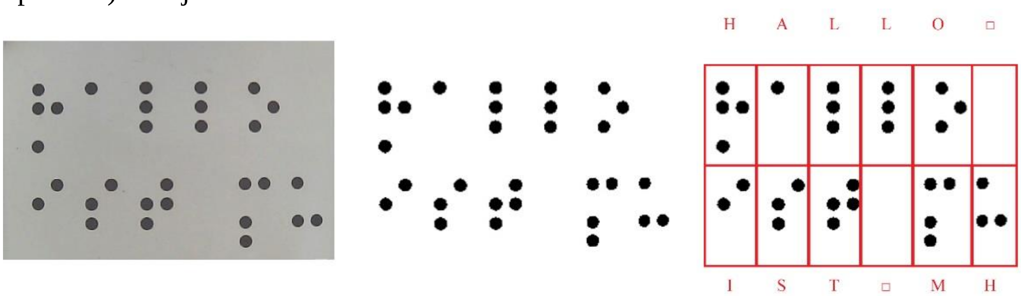


Fig. 4: Steps of image analysis : Left original colored image / middle: binarisation process after thresholding algorithm / right: pattern recognition (circle draw)

Figure 4 shows the original image (left), its binarized pattern by the Otsu's clustering-based image thresholding method (middle). The dots are grouped into the particular characters (right). This is achieved by measuring the distances between the dots and grouping them by vertical and horizontal segmentation. After identification of the individual groups and comparison with a Braille library, the characters are decoded.

3.2 Smart Code: Area recognition

Problem characterisation: In the following, we will demonstrate recognition algorithms that use a QR code as an example. The geometric form of the Finder pattern of the QR code is the same as in our smart code. By using a common static code such as the QR code, we want to provide an easier understanding of the following steps. Area recognition in a smart code involves the recognition of position markers (finder pattern) and the retrieval of data (static) and sensor (dynamic) areas.

Typically, the image of the camera captured the code is shaded, partial, or blurred. The code itself as well as the distance of single dots to each other or their patterns are distorted in both directions. All these erroneous parameters must be corrected before the code can be decrypted. High demands are placed on these corrections, especially with regard to robustness and reliability. Various suitable methods and algorithms have to be applied for the individual steps of error correction: e.g. the Circle Hough Transformation (geometric correction) described above or the Canny Algorithm (background suppression, edge detection and edge correction). These and other methods are available in program libraries and must be integrated operatively during programming.

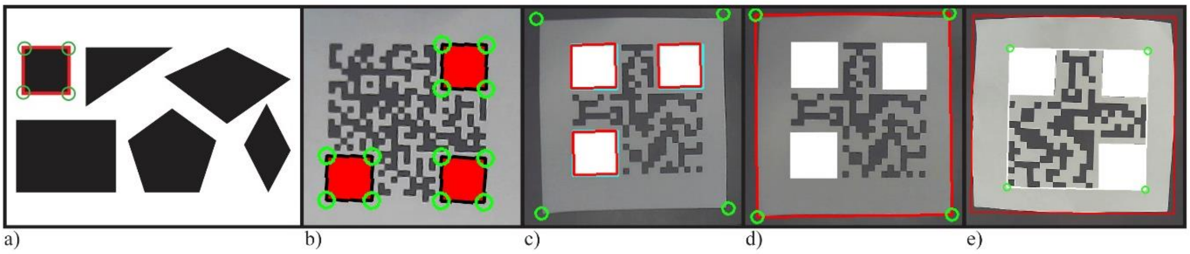


Fig. 5 Position marker recognition by geometric shaping

By recognition of geometric patterns, the individual position markers can be identified and therefore the data area as well as the sensor areas of a dynamic smart code can be located. This is particularly demonstrated in the prototype of the smart code in Figure 6. Based on the relative distances of the position markers distortions of the code can be balanced out. If the code is rotated, this will also be corrected. The square bars below and above the position markers identify the smart code as a code differing from a similar random pattern. In addition, the dynamic range for sensitive colours is identified.

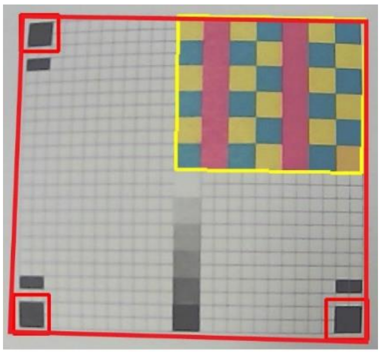


Fig. 6 Smart code prototype

3.3 Dynamic Data: Colour recognition

The following section deals with the evaluation and identification of the dynamic area (printable sensors) of the smart code. The respective Smart Materials are filled into the printer's cartridges instead of cyan, magenta and yellow. These three cartridges will contain a photochrome ink (photosensitive), pH-chrome ink (acid sensitive) and hydrochrome ink (humidity/ water sensitive). The black ink cartridge contains regular ink to prints all the static information of the code, including the finder pattern. The colourseparated areas (by delimiting the colour ranges), such as the yellow squares, three magenta lines and the cyan squares represent the smart materials and their cartridges (Fig.9).

By detecting the dynamic area of the smart code, the colour values - of a specific position - inside of the dynamic code can be analysed. The status (degree of the contamination) of the individual smart material can be interpreted by means of the colour information. The aim is to identify the current colour values of each smart material and compare them with the measured colour reference before. This way, it is possible to refer the specific colour values with the dimension of the contamination. In this context, as well all RGB colour information of the background can also be used for colour correction of the dynamic code in order to get finally the true colour information. However, the colours in figure 7 show significant deviations, due to the camera detection. This indicates the necessity of a calibration of the colour values of the camera. This is planned to be realised in one of the next project steps by application of a neural network.

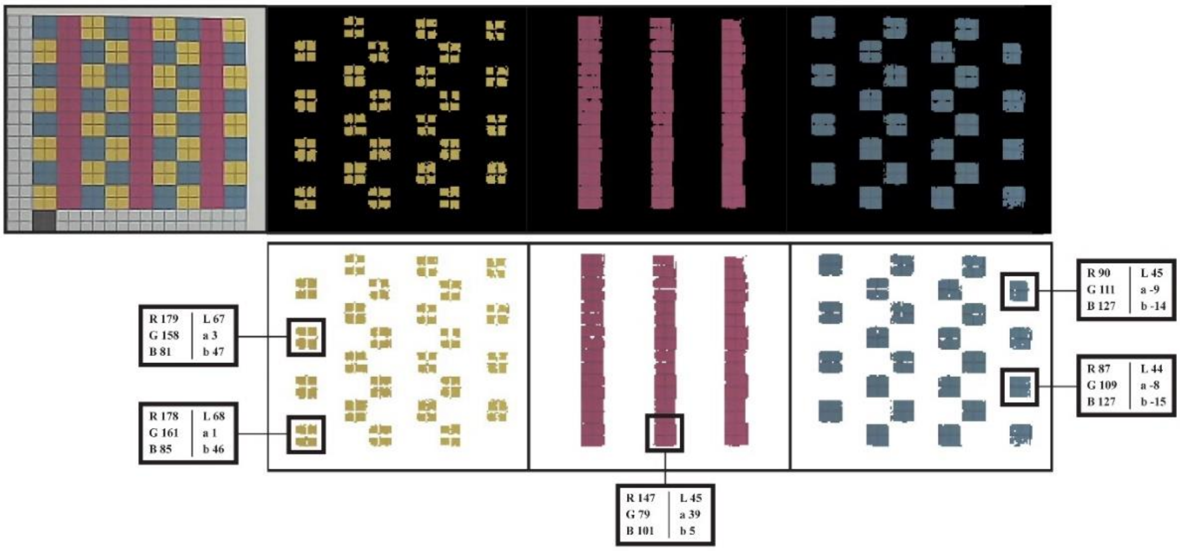


Fig. 7 Color detection and separation

The photochromic surface in figure 8 can be analysed by measurement of the RGB and CIEL*a*b values or alternative by RGB greyscale and L* values. Therefore it is possible to store information about critical deviations in a range from 0 (white) to 255 (black) in RGB or 0 (black) to 100 (white) in L*. A smart device can identify the RGB or CIEL*a*b* values of the different colour shades (smart dots) and compare them with the initial RGB or CIEL*a*b* values and set the colour difference (ΔE) into correlation with the irradiated quantity of light (photons) of the contamination.

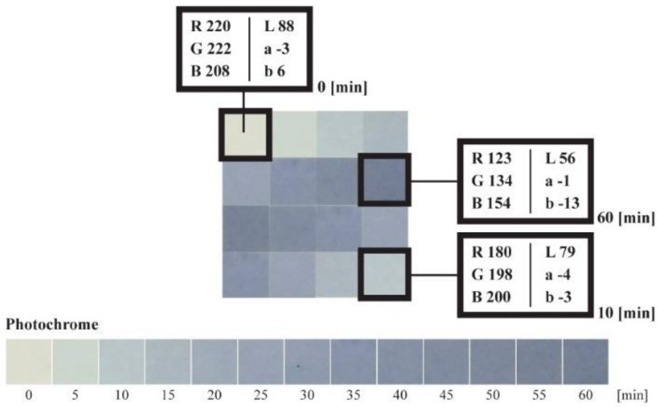


Fig. 8 Smart Material: Photochromic Ink

Table 1 shows the measurement data of the photochromic surfaces. A non-contaminated surface can be defined with an (R, G, B) tuple of (220, 221, 208) and a maximum exposed surface (60 minutes or longer UV exposition) can be described as a tuple of (121, 129, 150); 10 minutes exposition RGB is (174, 182, 188). The data of Table 2 allow deducing on the UV contamination of a product. As well, areas can be grouped in order to classify the quality of a product, e.g. good, moderate, and poor. Another application may be the indication of the entire UV intensity during the crosslinking process of UV printing ink or UV varnish or at crosslinking of UV adhesives.

Min of UV exposure	0	5	10	15	20	25	30	35	40	45	50	55	60
R	220	207	174	189	163	156	149	140	137	131	128	126	121
G	221	214	182	197	172	163	157	148	145	139	136	133	129
B	208	205	188	199	183	181	176	168	166	157	155	150	150
L*	88	85	79	74	70	68	65	61	60	58	57	54	54
a*	-2	-5	-4	-3	-2	-1	-2	-1	-1	-1	-1	-1	-1
b*	7	4	-2	-5	-8	-11	-12	-13	-11	-12	-12	-11	-13
ΔE	12.8	6.1	5.9	5.7	22.3	26.1	2	4.5	1	1.7	3.3	1.4	

Table 2 Measurement data of photochromic surfaces when UV exposed

3.4 Calibration

In the following, the calibration of the camera is discussed. The aim was to control the exposure and the white balance with a kind of colour checker function. The defined grey tones in figure 9 were integrated into the smart code and can be used for the calibration process. By capturing the smart code by using a code reader, the small areas within the grey areas (5 x 5 pixels) are captured in real time. Thus, the grey values and also the black values (Braille dots) in the data area are used to realize a correct exposure and white balance.

This was realized as described below:
First, the grey fields and their respective RGB colour values were measured and an average value determined (actual state). Next, the respective RGB averages were compared with those of the reference table (target state) and the respective deviation ΔE was calculated. The deviation ΔE was used as a colour check function to control the OpenCV exposure and white balance functions. The process was repeated as long until an acceptable exposure and white balance was achieved.

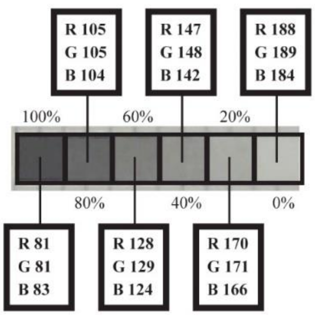


Fig. 5 Correct exposure and white balance using the colour checker function

Distortion manifestations such as barrel or fish eye effects have also to be corrected. The distortion matrix can be determined and corrected by using the asymmetric circle pattern resulting from the data range (Braille dots). Different lighting conditions - under the influence of light and shadow - have a significant effect to build a robust colour recognition system while doing colour interpretation.

4. Conclusions

It is possible to generate, print and read out a dot matrix code containing dynamic fields of sensitive inks with the camera of a smartphone. Furthermore, the colour information of the dynamic areas can be interpreted and correlated with corresponding contaminations. In laboratory scale, individual steps were shown to develop a smart code and to read it with a smartphone camera. To what extent the developed methods prove to be robust and to what extent the degree of several simultaneous contamination with the methods can be referenced and reproducibly determined must be demonstrated in comprehensive laboratory tests, whereby the developed process procedures must also be evaluated and readjusted. As soon as this will be verified, the data flow and the server operations can be designed and implemented in the IoT.

Here, some key steps are shown to develop a multifunctional sensor that works without its own power supply. The reading of the sensor information is possible by means of consumer devices and the interpretation of the information does a server within Internet of Things. Today, low availability of suitable, irreversible, sensitive dyes, which are inkjet print-able is problematically, yet. However, it should be supposed that this bottleneck quickly will be overcome as soon as these multifunctional, current less sensors have been developed beyond the laboratory scale and a market potential is identified.

References

Ashton, K. (2009). That 'Internet of Things' Thing; In the real world, things matter more than ideas. Hauppauge, New York, USA: RFID JOURNAL LLC. Retrieved from: www.rfidjournal.com /articles/view?4986. 06.03.2019

Bilgin, M. and Backhaus, J. (2017, a). Smart Packages by Means of Intelligent Codes. Advances in Printing and Media Technology. Advances Vol.44. Vol. XLIV(IV). Fribourg, Swiss: iarigai. ISBN 978-3-9870704-1-9. ISSN 2409-4021. p.89-96

Bilgin, M. and Backhaus, J. (2018). Development of a unidirectional switchable Photochromic Ink for Smart Packaging. Advances in Printing and Media Technology. Advances Vol.45 (2018). (Online) Vol. XLV(V), iarigai, Warsaw, Poland. ISBN 978-3-948039-00-4. ISSN 2409-4021. P.55-63

Hough V, Paul C. (1962). United States Patent 3069654, Application Number: US1771560A, Publication Date: 12/18/1962

International Standard ISO/IEC 10646 – Unicode. (2010). ISO 11548-1:2001. Communication aids for blind persons – identifiers, names and assignation to coded character sets for 8-dot Braille characters – Part 1: General guidelines for Braille identifiers and shift marks. Retrieved from: www. unicode.org/L2/L2010/10038-fcd10646-main.pdf. 06.03.2019.

OpenCV. (2019). About OpenCV. Retrieved from: www.opencv.org/about.html. 06.03.2019.

Pascolini D. and Mariotti SP. (2012). Global estimates of visual impairment. British Journal of Ophthalmology (BJO). London, UK: BMJ Publishing Group. May, 96(5):614-8. doi: 10.1136/bjophthalmol-2011-300539. p.1-5.

Yuen, H. K.; Princen, John; Illingworth, John; Kittler, Josef. (1990). Volume 8, Issue 1, February, Amsterdam, Netherlands: Image Vision Computer. DOI:10.1016/0262-8856(90)90059-E. p.71-77