Impact of sensor aging on iris recognition

Anonymous IJCB 2014 submission

Abstract

Similar to the impact of aging on human beings, digital image sensors develop aging effects over time. Since these imager's aging effects (commonly denoted as pixel defects) leave marks in the captured images, it is not clear whether this affects the accuracy of iris recognition algorithms. This paper proposes a method to investigate the influence of sensor aging on iris recognition by simulative aging of an iris test database. A pixel model is introduced and an aging algorithm is discussed to create a test database. To establish practical relevance, the simulation parameters are obtained from the observed 4-year-aging effects of a real iris scanner.

1. Introduction

The aging process starts immediately after being given birth. At first, aging's aspects are considered to be rather positive. One grows up, gets stronger and learns new things. At a certain point, aging starts to reveal several drawbacks. The skin gets wrinkled, joints and bones start to ache and also our vision starts to degenerate. In either case, aging changes parts of our body. Although human beings change, others are still able to recognize people they knew before, even when they haven't met in a long time and therefore aging changed their appearance significantly. Biometric systems aim to identify human beings by analysing their biometric samples, i.e. the face, the fingerprints or iris [27]. These systems operate in two steps. In the enrollment process, a subject's biometric samples are registered for the first time and stored in a database. Later, in the authentication procedure, another sample of a subject's biometric samples is taken and compared to the one stored in the database to verify the subject's identity.

Since there may be years or decades between the enrollment and the authentication, one strives to use biometric samples which are stable over time. The iris has widely been assumed to be stable, although it is currently topic of intensive discussion if and which iris-related information eventually changes in a human's aging process. Some researchers claim that iris-related information is stable or relatively stable [4, 5, 13, 20], while others observe significant changes over time [3, 9, 10, 23, 24]. Researchers mostly conclude these age-dependent changes in iris texture by observing changes in a system's iris-recognition rate. It is not clear, if observed changes in an algorithms' behaviour are caused by the aging of the tested subject, which is commonly assumed, aging of the recognition system itself or if this is caused by other unspecified factors.

This paper proposes a method to investitage the influence of aging effects of digital image sensors, which are crucial components of the recognition system. As a human being's vision often gets weaker with age, pixels start to get defective in image sensors over the years as well. This influences the quality - and therefore the appearance - of the captured images. Similar to being able to estimate a human being's age by his appearance, pixel defects observed in the captured images can reveal the age of the corresponding imager. This is used in image forensics to approximate the capturing date of an image [11]. However, since the aging effects of the sensor can be traced by evaluation of its output, namely the images, this implies that the capturing process is not time invariant. Hence two images, which capture the same (unchanged) scene, but were taken at significantly different points in time, may differ due to developed sensor defects in the meantime.

There might arise problems due to sensor aging, because iris recognition systems operate by comparing current samples to earlier taken ones stored in a database. Special cameras with digital image sensors are commonly used to capture the iris features in both cases, the enrollment and the authentication process. It is not clear yet, whether the aging of the sensor - and thus the development of aging effects between taking two samples - influence the accuracy of iris recognition algorithms. To investigate this issue, one would need to have identical data captured at at least two significantly different points in time. It is practically impossible to establish identical conditions for both shots. Furthermore, as mentioned, it is not possible to distinguish between the influence of the subject's and the sensor's aging on iris recognition algorithms. For this reason one cannot capture test data to investigate the sensor's or the subject's aging in an isolated manner physically. There is no way to explicitely examine the iris texture's aging at all. However, it is possible to investigate the sensor aging's influence on the iris recognition rate independent from the subject's aging.

We proposes a method to investigate the influence of a sensor's aging on the recognition rate. This is done by generating test data based on an the IITD iris database [22] (TODO: Check if all captured by one sensor) by a simulated aging process. The practical relevance of this simulation is ensured by retrieving the simulation parameters from two iris databases 'NAME' [28], where the same subjects were captured by the same sensor in 2009 and 2013. A method to retrieve the simulation parameters from this data is introduced in section 3.1. Using these simulation parameters, a pixel model and method to age images by generating images with hot and stuck pixels is discussed in 3. This test database is then used to test six implementations of iris recognition algorithms available in the University of Salzburg Iris Toolkit (USIT) [25] for influence of sensor aging effects upon the equal error rate (EER), refer section 4. Finally we discuss the results and limitations of our tests (see section 5) and interpret them to give a statement about the influence of sensor aging on iris recognition algorithms in section 6.

2. Sensor aging and pixel model

An image sensor is a two dimensional array of photosensitive cells called pixels. The purpose of an image sensor is to convert the incoming light to its digital representation, which is commonly denoted as the image [16]. In theory, each of these pixels has the same rectangular shape and a unified photo-response, meaning all pixels should produce the same output value on unified incoming light. Due to imperfections in the manufacturing process, this is not the case in practice, hence there are pixels which are more sensitive than others [15]. Besides that, some of a sensor's pixels might change their initial photo-response characteristic over the lifetime of a sensor. We denote this defect-development over time as *sensor aging*. Related work shows that such pixel defects caused by aging are equally distributed and independent throughout the sensor. Furthermore, it is well known that the development of pixel defects linked to material-related faults is an increasing function of time [1, 6–8, 11, 12].

The most common defects occuring over time are partially and fully stuck pixels, illustrated in fig. 1. Partially-stuck pixels tend to have a moderate offset compared to their neighbours, but still respond to the incident light. Stuck pixels produce a completely light-independent output value and are often (but necessarily) saturated [11]. As soon as a pixel is defective, it remains defective over the rest of the sensor's lifetime [1], regardless which type of defect it suffers from. So one can obtain the important property that if a pixel is *once defective, it remains defective*. This is shown in fig. 2.

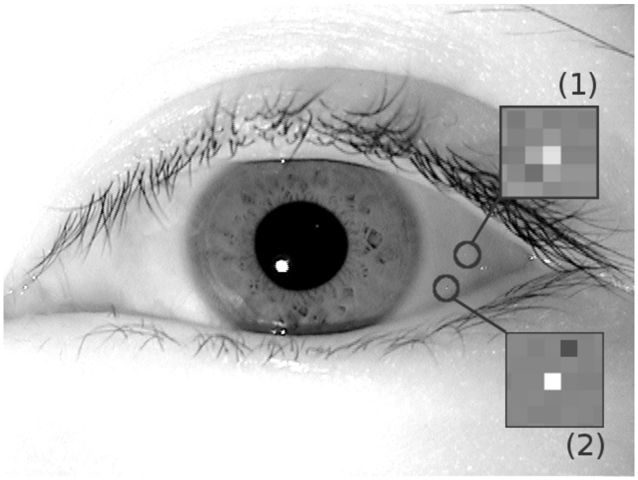

Figure 1. A *partially-stuck or hot pixel* (1) and two *stuck pixels* (2) in an iris image. Contrast has been enhanced in the close-ups for visualisation. Note that the partially-stuck pixel still corresponds to the incoming light, but is significantly brighter compared to its neighbours, while the stuck pixel are light-independent and yield to a constant output.

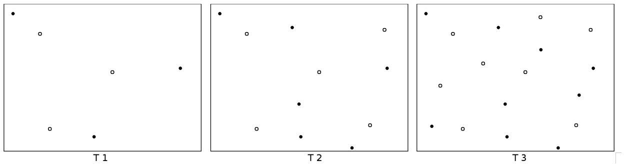

Figure 2. Illustration of uniformly distributed and independent locations of partially-stuck (circle) and stuck (filled circle) pixels in images taken at 3 significantly different points in time T1, T2, T3.

The occurrence of new pixel defects can be modelled by a poisson process, because the number of pixel defects is increasing with time and defect locations are uniformly distributed and independent. Already known defects remain constant in respect to their location and type [8, 11]. To run a simulation, a model for calculating the pixel values of an image y has to be defined. There are several pixel output models around, which consider the incoming light and the impact of pixel defects on the raw output of a sensor. We denote $w, h \in \mathbb{Z}$ as the rows and columns of the sensor and start by adopting the model proposed in [11]:

$$\begin{gathered} Y = I + I \circ K + \tau D + c + \Theta \\ with \quad Y, I, K, D, C, \Theta \in \mathbb{R}^{w \times h}; \tau \in \mathbb{R} \end{gathered} \tag{1}$$

where Y is the sensor output, commonly denoted as image, I the intensity of incoming light, $I \circ K$ the photo-response non-uniformity PRNU, τD the dark current (with τ being a multiplicative-factor representing exposure settings, sensor temperature, ...), C a light-independent offset and Θ some modeling noise. Since all pixels are independent [7, 11] and all operations element-wise, we denote the

matrix-elements $y_{x,y} \in Y$ as $y \in Y$ for simplicity reasons. The same applies to $i \in I$, $k \in K$, $d \in D$, $c \in C$ and $\theta \in \Theta$.
Since we're interested in simulating the aging effects of a specific sensor, all age-independent defects, which are e.g. due to production process, can be eliminated. Hence the PRNU, which corresponds to the non-uniformity of pixel-dimensions, can be omitted. As we're interested in reproducable tests, this means environmental influence (e.g. temperature) and modelling noise should be minimized. They can be eliminated completely in a simulation, thus we set $k = \theta = 0$. Also the same exposure settings have to be used for the sensor in every test, therefore we have $\tau = const$ and set $\tau = 1$ for simplicity. [1, 11, 14, 15] suggest that the dark current is very weak with short exposure, which is necessary for applications in biometric systems to avoid motion blur. Under these considerations, a fairly simple model, considering only aging-relevant defects, remains:

$$y = i + d + c \quad with \quad y, i, c \in \mathbb{R} \tag{2}$$

Common defect types that develop over time as the sensor ages are stuck and hot pixels, where the definitions are quite contrary in literature (e.g. in [1, 8, 11]). If the dark current d of a pixel is extremely high, this is often denoted as a hot pixel. If the offset c is high, this results in a saturated pixel and is called a stuck pixel according to [11]. [8], however, suggests that a stuck pixel is not necessarily saturated but can obtain any value within the sensor output's universe. For the reason of non-uniform definitions we define the following models for defective pixels:

$$\begin{aligned} y &= c & (3) \\ y &= i + d & (4) \end{aligned}$$

where the defect type in equ. 3 is light-independent (thus referred as a *stuck pixel*) and the one in equ. 4 adds an offset to the incident light and is commonly referred as either a partially-stuck or hot pixel. The dark current (which is the cause for hot pixels) depends on exposure time and temperature, which are both kept constant this experiment. Thus also the dark current is constant and therefore there is no difference between a hot and a partially stuck pixel for this setup. We will denote this effect as a *partially-stuck pixel*. In conclusion (and considering 8 bit grayscale images), the following pixel model is definied:

$$Y_{x,y} = \begin{cases} C_{x,y} & \text{if } c_{x,y} \neq 0; \\ I_{x,y} + D_{x,y} & \text{otherwise.} \end{cases} \tag{5}$$
$$\text{with} \quad Y, C, I, D \in (\mathbb{Z} : [0; 255])^{w \times h}$$

where $Y_{x,y}$-values are clipped at 0 and 255 respectively if interval borders are exceeded. This means they are able to saturate.

3. Simulated sensor aging

For an ideal sensor, the defect matrices C and D are zero matrices at a specific point in time T_0. As pixel defects occur at a constant rate, this can be modelled by a poisson process [11]. Hence the number of stuck pixels and partially stuck pixels, denoted as n_s and n_{ps} respectively, at T_i can be calculated as

$$\begin{aligned} n_s(t) &= (T_i - T_0)\lambda_s & (6) \\ n_{ps}(t) &= (T_i - T_0)\lambda_{ps} & (7) \end{aligned}$$

where λ_s and λ_{ps} are the rates at which the particular defect types occur. Due to defects being independent of each other, the locations of the defects among a 2D sensor array can be modelled by a uniform distribution. Thus the defect location (x, y) can be obtained using random numbers r. We propose the following procedure to obtain the position $s_k \in w \times h$ of the k-th pixel defect:

$$s_k(x, y) = (\left\lfloor \frac{r}{w} \right\rfloor, r \mod w)) \quad \text{with} \quad r \in [0 : w \cdot h] \tag{8}$$

Depending on the k-th defect being a stuck or partially stuck pixel, the values of C and D according to the model defined in equ. 5 have to be set. We denote a_s as the maximum amplitude of a stuck pixel and a_{ps} the maximum amplitude of a partially stuck pixel. Let $r_a \in \mathbb{R} : [0; 1]$ be a uniformly distributed random number. Then we either have

$$\begin{aligned} C_{s_k} &= r_a \cdot a_s \quad \text{for stuck pixel at } s_k \text{ or} & (9) \\ D_{s_k} &= r_a \cdot a_{ps} \quad \text{for partially stuck pixel at } s_k & (10) \end{aligned}$$

These definitions together with the pixel model introduced in equ. 5 can be used to simulatively add aging-related sensor defects to an existing image Y_{T_0} captured at time T_0. One might argue that Y_{T_0} already contains pixel defects since the used sensor might bear defects at T_0 already. Since we are only interested in investigating if something changes over a period of time, it does not matter which time frame we observe. This means, already contained defects in Y_{T_0} do not influence the outcome of the experiment. Using the pixel model, sensor defects corresponding to a sensor's physical status at a specific time T_i can be embedded in the image Y_{T_0}. The resulting image Y_{T_i}, denoted as *aged image*, can be interpreted as an image capturing the completely same scene as in T_0, but revealing the effects of sensor aging over a period of time $\Delta t = T_i - T_0$. Because sensor defects change over time, we define a sequence of defect matrices, which represent the state of aging at a specific point in time T_i. We denote these sequences corresponding to sample points $T_0 \dots T_m$ as

$$\begin{aligned} &(D_i)_{i=0..m} \text{ and } (C_i)_{i=0..m} \\ \text{with} \quad &C_i, D_i \in (\mathbb{Z} : [0; 255])^{w \times h} \end{aligned} \tag{11}$$

With these definitions the influence of the physical state of the sensor at a considered sample point T_i can be embedded in an image simulatively. The following algorithm is proposed to compute the sequence of aged images $Y_{T_1} \dots Y_{T_m}$ from a source image Y_{T_0}:

```
procedure AgedImageSequence($Y_{T_0}$)
    for $i = 1 \dots m$ do
        $\Delta n_s \leftarrow n_s(T_i - T_0) - n_s(T_{i-1} - T_0)$
        $\Delta n_{ps} \leftarrow n_{ps}(T_i - T_0) - n_{ps}(T_{i-1} - T_0)$
        $D_i = D_{i-1}$
        $C_i = C_{i-1}$
        for $k = 1 \dots \Delta n_s$ do
            $r_a \leftarrow$ random in $[0; 1]$
            $s_k \leftarrow$ random in $w \times h$ (equ. 8)
            $C_{s_k} \leftarrow r_a \cdot a_s$ (equ. 9)
        end for
        for $k = 1 \dots \Delta n_{ps}$ do
            $r_a \leftarrow$ random in $[0; 1]$
            $s_k \leftarrow$ random in $w \times h$ (equ. 8)
            $D_{s_k} \leftarrow r_a \cdot a_{ps}$ (equ. 10)
        end for
        $Y_{T_{i_{x,y}}} = \begin{cases} C_{i_{x,y}} & \text{if } C_{i_{x,y}} \neq 0; \\ Y_{T_{0_{x,y}}} + D_{i_{x,y}} & \text{otherwise.} \end{cases}$
    end for
    return $(Y_{T_i})_{i=1 \dots m}$
end procedure
```

Basically for each point in time the number of defects-to-add $\Delta n_s, \Delta n_{ps}$ are calculated. The defect matrices D and C are computed recursively. This is necessary to satisfy the *once defective, always defective* condition. By recursive calculation the earlier developed defects are maintained over virtual age. As soon as the defects for the i-th aging step are defined, they are used to compute an aged image Y_{T_i} using the pixel model defined in equ. 5. Instead of the incident light I, the base image Y_{T_0} can be used due to the previously given arguments. At the end we obtain a sequence $(Y_{T_i})_{i=1 \dots m}$ Each pixel sequence y_i represents completely the same incident light as in Y_{T_0} but also bears the information of the age-dependent defects of a sensor for a specific point in time T_i.

3.1. Parameter retrieval from iris databases

For the discussed simulative aging process, the defect's growth rate and amplitudes have to be defined. In laboratory set-ups hot and *partially stuck pixels* are usually identified by dark calibration tests (i.e. $I = 0$) [8]. To the best of the authors' knowledge, there is no suitable laboratory-captured data set available for iris imagers. There are databases [3, 28] available, which were captured using the same equipment at two significantly capturing dates for the purpose of investigating iris texture aging. In the following we propose a method to retrieve the growth rate and amplitudes of partially and fully stuck pixels from such databases.

For the pixel model defined in equ. 5, the growth rate and amplitudes of partially and fully stuck pixels have to be detected. As discussed, this type of defect is usually detected from a single image when the incident light I is simply set to an uniform and known value, i.e. $I = 0$. Hence only the defects $\Xi = D + C = Y - I$ remain. Because in our database I is unknown, a statistical approach has to be used. We denote $Y_0 \dots Y_K$ a sequence of K images taken in a very short period of time. We know that a sensor defect has to be contained in each of this image and is (according to equ. 5) independent of I. A *stuck pixel* obtains the same value in each image in this sequence. Hence a pixel $y \in Y$ at position (x, y) is identified as being stuck iff

$$y_0 = y_1 = \dots = y_K \tag{12}$$

To additionally find partially stuck pixels, we compute a pixel's mean $\bar{y}$ from the sequence of images:

$$\bar{y} = \frac{1}{K} \sum_{k=1}^{K} y_k \tag{13}$$

$$\bar{y} = \frac{1}{K} \sum_{k=1}^{K} (y_k + d + \theta_k) \tag{14}$$

$$\bar{y} = d + \frac{1}{K} \sum_{k=1}^{K} (y_k) \tag{15}$$

Equ. 14 is obtained by using the pixel model of equ. 5 (without c, because stuck pixels fulfilling equ. 12 are ruled out already) and adding some modelling noise θ. Since d is a defect and therefore contained in every one of the sequences images, it is independent of k. The mean of a uniformly distributed modelling noise θ cancels out for sufficiently large K. So we get equ. 15.

The mean $\bar{y}$ at a certain location has the same value as if taken the median over several $\bar{y}_i$ in its local neighbourhood with distance q. This holds if the neighbouring pixels $\bar{Y}_{x \pm q, y \pm q}$ had similar pixel values in the original images Y_i. Therefore we have the identity $med(y, q) = \bar{y}$, even if one (but less than half) of the $\bar{y}i$ in the neighbourhood is significantly different. Hence sparse pixel defects, which contribute to the mean, disappear when applying median filtering on the mean image's pixels $\bar{y}$ in a local neighbourhood. Considering only parts of the mean image $\bar{Y}$, where

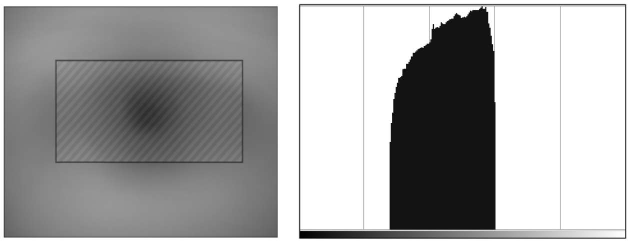

Figure 3. The mean image of $Y_1 \dots Y_k$ (left) has correlated data in the image's centre because of the iris usually being there. This (marked) region is not considered when detecting sensor defects. The histogram (right) of the uncorrelated regions is indeed similar to uniform distribution

mostly unified brightness and texture is present in $Y_1 \dots Y_K$ we compute the estimator $\hat{d}$ for d as

$$med(\bar{y}, q) = \hat{d} + \frac{1}{K}\sum_{k=1}^{K}(y_k) \qquad (16)$$

$$\hat{d} = \frac{1}{K}\sum_{k=1}^{K} y_k - med(\bar{y}, q) \qquad (17)$$

Amongst the partially stuck pixel information, also information about the PNRU is also contained in $\hat{D}$. Since PNRU is caused by imperfections in the manufacturing process, it is likely to be normally distributed which reflects in the logarithmic histogram of $\hat{D}$, as shown in fig. 4. The size q has to be chosen large enough to minimize the influence of PNRU on the median $med(\bar{y}, q)$, which is the case if a normal distribution within the neighbourhood is observed.

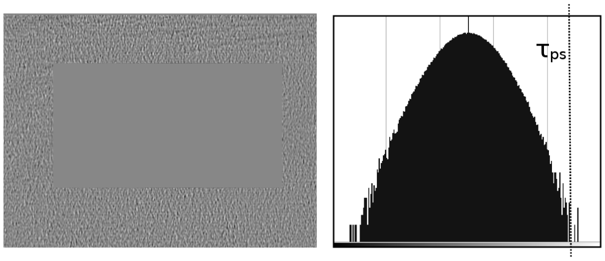

Figure 4. Matrix D (left, enhanced for visualisation) is calculated by equ. 17. The logarithmic histogram (right) of the uncorrelated regions shows a normal distribution, which is due to PNRU. The decision threshold τ_{ps} is so that only outliers are declared as partially stuck pixels

We declare a pixel to be partially stuck if $\hat{d}$ is an outlier and therefore higher sensivitiy is unlikely to correspond to PNRU. This is decied by checking for a certain threshold $\hat{d} > \tau_{ps}$. The decision threshold is chosen, i.e. manually, to just get approximately same amount of outliers in T_1 and T_2. We exploit the *once defective, always defective*-property to guarantee correct results, even if τ_{ps} has been slightly off. We know that a sensor's partially stuck pixels from T_1 also have to be present in T_2. This is illustrated in fig. 5. The number of defects n_{match}, which are present at the same location s_k in the images captured at T_1 and T_2 can be determined. Considering the number of defects n_1 in the image captured at T_1, the correction factor γ of the algorithm can be calculated as

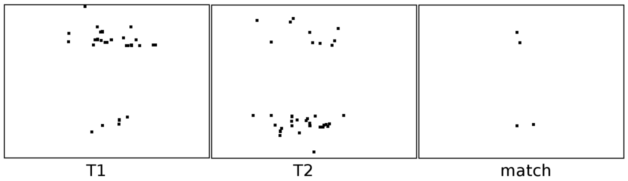

Figure 5. Locations of partially-stuck pixel candidates. Correctly classified pixel defects from T_1 are contained in T_2 as well. All other detected defects in T_1 can be interpreted as misclassification, since they violate the *once defective, always defective*-condition.

$$\gamma = \frac{n_{match}}{n_1} \qquad (18)$$

Assuming that for T_1 and T_2 the same error is made, the observed increase of defects can be corrected with γ. Taking into account the size of the sensor $w \cdot h$, we retrieve the simulation parameters by the following relations:

$$\begin{aligned}
\lambda_{ps} &= \gamma \frac{n_2 - n_1}{(T_2 - T_1)\cdot(w \cdot h)} && (19)\\
\lambda_{s} &= \gamma \frac{n_{s2} - n_{s1}}{(T_2 - T_1)\cdot(w \cdot h)} && (20)\\
a_{ps} &= max(\hat{D}_{s_k}) && (21)\\
a_{s} &:= 255 && (22)
\end{aligned}$$

4. Experimental setup

The proposed method is used to retrieve simulation parameters from two databases, which cover the same test subjects and were captured using the same sensor in 2009 and 2013.

TODO Luca:

- I'd call this paragraph 'Relevance of the used database'
- Introduce database (“release note”)
- Argue relevance of data base (one sensor only) ...
- Mention methods of prove why they are really from the same sensor (pls use BibTex-References)

We used the proposed method in section 3.1 to retrieve the growth rate and amplitudes of pixel defects for the H100 sensor. By chosing $q = 3$ and appropriate τ_{ps} as decision thresholds, $n_{ps}(2009) = 37$ and $n_{ps}(2013) = 38$ candidates for partially-stuck pixels were found in 2009 and 2013 respectively (see fig. 5). Because there where $n_{match} = 4$ matching pixels, the correction factor is set to $\gamma = 0.1081$. Using these values, a estimated growth rate $\lambda_{ps} = 0.129$ can be determined. The growth rates are measured in number of defects per Megapixel per year. The retrieved values (refer table 4) correspond with the findings of other researchers, refer [7, 18]. Dudas suggests in [6] that real stuck pixels are never detected in field, because they are easy to detect at fabrication time and get corrected or may correspond to partially stuck pixels with extremely high offset. However, we modelled actual stuck pixels at a moderate growth rate (refer table. 4) to investigate what happens if stuck pixels should occur. Furthermore, we modelled other sensors with growth rates retrieved from literature and pushed the aging parameters to mutiples of the real values. To get the highest possible impact, the largest growth rates for the specific sensor were taken. A summary of all used parameters in this simulation is given in table 4.

Parameters	h100	APS [18]	CCD [18]
$\lambda_{ps}\left[\frac{\text{defects}}{\text{year}\cdot 10^6\text{pixels}}\right]$	0.129	0.743	0.569
λ_s	0	0	0
a_s	255	255	255
a_{ps}	3	?	?

$1MP := 10^6 pixels$

Table 1. Simulation parameters.

Based on these parameters, a number of virtual sensors are defined in table 4, where sensor A corresponds to the H100. They are designed in a way that all possible combinations of partially-stuck and stuck pixels are covered. Sensors F,G and H show amplitudes and growth rates which are very unlikely to be observed in practice, but should demonstrate the influence of heavy aging effects.

Sensor	λ_{ps}	λ_s	a_{ps}
A	λ_{h100}	0	$a_{ps_{h100}}$
B	λ_{h100}	λ_{APS}	$a_{ps_{h100}}$
C	λ_{APS}	0	$a_{ps_{h100}}$
D	λ_{CCD}	0	$a_{ps_{h100}}$
E	0	λ_{APS}	–
F	$4\lambda_{h100}$	0	$2a_{ps_{h100}}$
G	$4\lambda_{h100}$	0	100
H	$8\lambda_{h100}$	$5\lambda_{APS}$	100

Table 2. Virtual sensors for simulation

For each of these sensor models, an aged data set based on the IITD database [22] (TIFF, N=2240) over a time span of 96 years is calculated with a step width $\Delta t = 8$years. This is likely to cover sensor aging over a human being's entire lifetime. The impact of sensor aging is determined by evaluation of the Equal Error Rate EER (also known as Crossover Error Rate). In terms of this paper the EER is calculated by considering the left and right eye of a subject as seperate classes. By the structure of the database (using crossover evaluation), 8940 genuine and 5006400 impostor matches are the base for computing the EER.

Sensor A, which is a model of a real iris scanner, has been tested with all six implementations iris-signature algorithms of the USIT Framework to evaluate each algorithm's vulnerability to sensor aging. For all other virtual sensors, the algorithms of Ma *et al* [19] and Monro *et al* [21] were used. The segmentation was carried out by Weighted adaptive Hough and ellipsopolar transforms WAHET [29].

5. Results

For all six iris signature algorithms no tendency in accuracy change due to sensor aging is observed. This is concluded from investigating the EER on sensor A (refer fig. 6). There is, however, a significant variation in EER at the sample points for some algorithms. Thus defects caused by sensor aging indeed influence the retrieved iris signatures. Especially for the LogGabor-1D method [17], a high variation between sample points is observed. This indicates that this method is sensitive towards sensor aging. The algorithm of Ko *et al* [2] and the context based method [26] tend to be robust.

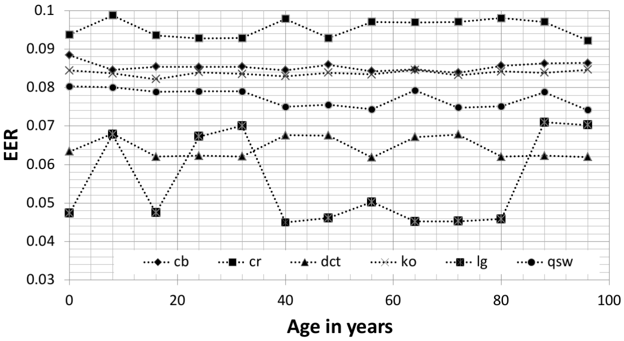

Figure 6. EER using Sensor A and WAHET segmentation

5.1. Limitation of the results

- Only one sensor tested (probably not representative)
- In-Camera fixing of sensor defects neglected (might influence development rate, but not so important, because also tested with multiples of the original rate)
- unsure whether noise reduction on in iris sensors
- But proved in JLEUNG that was off... so these rates probably representative
- segmentation or iris code?

6. Conclusion

We used a simulation to embed aging-related defects onto a sensors output. Practical relevance of the simulation has been established by retrieving simulation parameters from data, which has been captured by an iris scanner in a 4 years' difference. To retrieve these parameters, a method was proposed in section 3.1. By using these parameters and similar ones from related work, an experiment has been carried out. In this experiment the impact of sensor-aging on the accuracy of iris recognition systems was investigated.

Interpreting the results, we can say that the LogGabor-1D method is vulnerable to aging-related defects. The algorithm's of Ma et al, Monro at al and Rathgeb and Uhl's DCT based algorithm are rather stable, whereas Rathgeb and Uhl's context based algorithm and the algorithm of Ko at al is considered to be robust. However, no tendency to a overall decrease in recognition accuracy can be determined. In conclusion, this paper shows that some algorithms are more vulnerable to age-related defects than others, although no general tendency in accuracy can be observed.

References

[1] E. Amerasekera and F. Najm. *Failure Mechanisms in Semiconductor Devicess*. John Wiley & Sons, New York, 1997.

[2] A. Azizi and H. Pourreza. A novel and efficient method to extract features and vector creation in iris recognition system. In B. Mertsching, M. Hund, and Z. Aziz, editors, *KI 2009: Advances in Artificial Intelligence*, volume 5803 of *Lecture Notes in Computer Science*, pages 114–122. Springer Berlin Heidelberg, 2009.

[3] A. Czajka. Template ageing in iris recognition. In *The 6th International Conference on Bio-Inspired Systems and Signal Processing*, Barcelona, Spain, 2013.

[4] J. Daugman. Biometric personal identification system based on iris analysis, Mar. 1 1994. US Patent 5,291,560.

[5] J. Daugman and C. Downing. No change over time is shown in rankin et al. ”iris recognition failure over time: The effects of texture”. *Pattern Recogn.*, 46(2):609–610, Feb. 2013.

[6] J. Dudas. Characterization and avoidance of in-field defects in solid-state image sensors. Master's thesis, Simon Fraser University, Canada, 2008.

[7] J. Dudas, C. Jung, G. H. Chapman, Z. Koren, and I. Koren. Robust detection of defects in imaging arrays. In *SPIE-IS&T/ Vol. 6059 60590X-12*.

[8] J. Dudas, L. M. Wu, C. Jung, G. H. Chapman, Z. Koren, and I. Koren. Identification of in-field defect development in digital image sensors. In *SPIE-IS&T/ Vol. 6502 65020Y-1*.

[9] M. Fairhurst and M. Erbilek. Analysis of physical ageing effects in iris biometrics. *Computer Vision, IET*, 5(6):358–366, November 2011.

[10] S. Fenker, E. Ortiz, and K. Bowyer. Template aging phenomenon in iris recognition. *Access, IEEE*, 1:266–274, 2013.

[11] J. Fridrich. Sensor defects in digital image forensic. In H. T. Sencar and N. Memon, editors, *Digital Image Forensics*, pages 179–218. Springer New York, 2013.

[12] M. Goljan and J. Fridrich. Determining approximate age of digital images using sensor defects. In *SPIE Conference on Media Watermarking, Security, and Forensics*, 2011.

[13] Grother, P. and Matey, J.R. and Tabassi, E. and Quinn, G.W. and Chumakov, M. . Irex vi temporal stability of iris recognition accuracy. Technical report, 2013.

[14] G. Healey and R. Kondepudy. Radiometric ccd camera calibration and noise estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(3):267–276, Mar 1994.

[15] G. C. Holst. *CCD Arrays, Cameras, and Displays*. JCD Publishing & SPIE Pres., USA, 2nd edition, 1998.

[16] J. Janesick. *Scientific Charge-Coupled Devices*. SPI Press Monograph, 2001.

[17] A. T. Kahlil and F. E. M. Abou-Chadi. Generation of iris codes using 1d log-gabor filter. In *Computer Engineering and Systems (ICCES), 2010 International Conference on*, pages 329–336, Nov 2010.

[18] J. Leung. Measurement and analysis of defect development in digital imagers. Master's thesis, Simon Fraser University, Canada, 2011.

[19] L. Ma, T. Tan, Y. Wang, and D. Zhang. Personal identification based on iris texture analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(12):1519–1533, Dec 2003.

[20] D. M. Monro, S. Rakshit, and D. Zhang. Dct-based iris recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(4):586–595, Apr. 2007.

[21] D. M. Monro, S. Rakshit, and D. Zhang. Dct-based iris recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):586–595, April 2007.

[22] I. I. of Technology. Iit delhi iris database (version 1.0).

[23] D. M. Rankin, B. W. Scotney, P. J. Morrow, and B. K. Pierscionek. Iris recognition failure over time: The effects of texture. *Pattern Recogn.*, 45(1):145–150, Jan. 2012.

[24] D. M. Rankin, B. W. Scotney, P. J. Morrow, and B. K. Pierscionek. Discussion: Iris recognition-the need to recognise the iris as a dynamic biological system: Response to daugman and downing. *Pattern Recogn.*, 46(2):611–612, Feb. 2013.

[25] [27]. University of Salzburg Iris Toolkit v1.0.3, 2014.

[26] C. Rathgeb and A. Uhl. Context-based biometric key generation for iris. *Computer Vision, IET*, 5(6):389–397, November 2011.

[27] C. Rathgeb, A. Uhl, and P. Wild. *Iris Recognition: From Segmentation to Template Security*. Springer, New York, 2012.

[28] U. Sbg. Aged iris data, 2014.

[29] A. Uhl and P. Wild. Weighted adaptive hough and ellipsopolar transforms for real-time iris segmentation. In *Biometrics (ICB), 2012 5th IAPR International Conference on*, pages 283–290, March 2012.