

# Framework for Image Sensor Design Parameter Optimization for Pupil Detection

Gernot Fiala  
Institute of Technical Informatics  
Graz University of Technology  
Graz, Austria  
OS R&D SYS SW  
ams-OSRAM AG  
Premstaetten, Austria  
gernot.fiala@{tugraz.at, ams-osram.com}

Zhenyu Ye  
OS R&D SYS SW  
ams-OSRAM AG  
Premstaetten, Austria  
zhenyu.ye@ams-osram.com

Christian Steger  
Institute of Technical Informatics  
Graz University of Technology  
Graz, Austria  
steger@tugraz.at

**Abstract**—Machine vision systems (MVS) use image sensors to process and analyze image data. Depending on the application, the image sensor parameters are configured differently. However, some parameters are fixed for a specific product generation or product line. One of these parameters is the pixel pitch, the distance from one physical pixel to another. In this work, we introduce a framework, which allows to optimize design parameters of image sensors for pupil detection. We compare 2 different image sensor models with different pixel designs and generate images with different bit depths and resolutions. An evaluation of the design parameters is done with the generated images and a pupil detection algorithm. Furthermore, an existing pupil detection dataset is extended.

**Index Terms**—image sensor, design parameter optimization, pixel pitch, bit depth, pupil detection

## I. INTRODUCTION

Machine vision systems (MVS) are widely used in the industry or agriculture for quality control to monitor the production and sort out bad or defected products [1]–[6]. They are also used in the consumer market for augmented reality (AR) and virtual reality (VR) applications. The applications of these MVS are different and therefore, the camera parameters are set differently for the specific use cases. However, some parameters can not be changed after production of the sensors. Such parameters are the pixel design, the distance between physical pixels (pixel pitch), the physical pixel size or the resolution of the image sensors. In any case, it is desired to optimize the image sensor parameters for certain use cases.

The focus of this work is on pupil detection and optimizing design parameters for future image sensors for AR/VR applications. These future image sensors for AR/VR applications should be smart with in-sensor processing [7], [8]. This means, that algorithms are directly processed on the image sensor. With these smart image sensors, power consumption of the communication from the sensor to a host can be reduced and the host platform can be further optimized to save energy consumption [9]. Since there are only limited resources on the image sensor available, especially for processing units and memory, optimizations of the whole image processing system and of the image sensor design parameters are required.

Some of these parameters are image resolution, bit depths and pixel pitch. Optimizing these parameters can lead to a lower memory usage for in-sensor processing and lower the power consumption of the image processing system.

The main contributions of this work are:

- A framework to simulate image sensor design parameters based on 2 different image sensor models with different pixel designs, resolutions and bit depths. The output images are evaluated with an pupil detection algorithm.
- An extension of an existing pupil detection dataset with images generated by 2 different image sensor models with different pixel designs, resolutions of 100x100, 200x200, 300x300, 400x400 and 500x500 pixels and bit depths from 8 bits down to 1 bit. In total the dataset is extended with additional 31200 images.

This paper is organized as follows: Section II shows related work of simulation frameworks for image sensors and optimizations of image sensors or image processing systems. Section III explains the framework for design parameter optimization, shortly explains the used pupil detection algorithm and gives an overview of the extended dataset. Section IV presents the results and Section V concludes the paper.

## II. RELATED WORK

Machine vision systems are used in different environmental conditions and the camera parameters can influence the results of the image processing system. Therefore, adjustments of the camera parameters for the different use cases and environmental conditions are necessary. P. Sibendu et al. introduce in [10] a framework, CamTuner to automatically and dynamically adapt sensors based on changes of the environment. They automatically update camera parameters (contrast, brightness, sharpness and color saturation) and evaluate the influence of these parameters on the accuracy of face recognition, person detection and face detection. With their framework, the accuracy can be increased. Another simulation framework, described by T. Nürnberg et al. in [11], focuses on mandatory components of computational cameras. Such components are apertures, lenses, spectral filters and sensors. The framework

consists of implementations of camera, sensor and tracer and can generate ground truth data to evaluate the performance of an image processing system. Another approach, described by G. Fiala et al. in [12] is about optimizing image sensors for specific applications, namely pupil detection for AR/VR applications. They propose to reduce the bit depth of images for pupil detection. A reduced bit depth can reduce the memory consumption for in-sensor processing.

Additionally, the power consumption of the communication from the image sensor to a host can be reduced by compressing data. The image sensor data are compressed with a neural network into a transmission map described by Pinkham et al. in [13]. With this approach, energy of the communication from the sensor to the host can be saved. However, data decompression is needed and therefore, the overall energy consumption stays similar. Another method to optimize an image processing system is splitting deep neural networks (DNN) into 2 processing layers. The first layer (L1) is for in-sensor processing and the second layer (L2) is for edge processing, described by Pinkham et al. in [14]. They analyzed the performance and energy of MobileNetV3 [15] and ResNet-50 [16] by finding the optimal split location of the neural network. The neural network is processed at L1 and L2 with different processors and different caching strategies. Based on the neural network split, size and type of the processor, the system energy consumption and performance was analyzed. Smaller networks are more suitable for these two-processor systems, because the minimum energy location is shifted for smaller networks to a later layer.

In this work a simulation framework is introduced to optimize image sensor design parameters, namely pixel pitch, resolution and bit depth. 2 different image sensor software models are used to generate images with the named design parameters. The images are used to extend an existing pupil detection dataset and a pupil detection algorithm is used to evaluate the image sensor design parameters.

### III. METHODOLOGY

The main idea of this work is to optimize image sensor design parameters for a pupil detection use case for AR/VR applications. Since future sensors for AR/VR applications should be smart with in-sensor processing [7], [8] and images should be processed with limited resources, optimizations of the image processing system are required. Such optimizations are pixel pitch, resolution and bit depths. Therefore, novel sensor designs are required.

This work proposes a framework to optimize image sensor design parameters, namely pixel pitch, resolution and bit depths for AR/VR applications. The parameters are then evaluated with an pupil detection algorithm. An overview of the framework is shown in Fig. 1.

In this work, the rendered images with a resolution of 500x500 pixels from [12] are used as input images for the image sensor models. Both image sensor models are implemented in Matlab<sup>1</sup> and are proprietary, but the methods described in

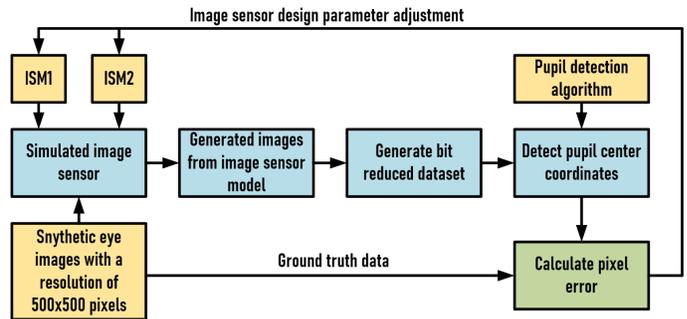


Fig. 1. Overview of the framework for image sensor design parameter optimization.

TABLE I  
VALUES FOR PIXEL PITCH USED FOR THE IMAGE SENSOR MODELS TO GENERATE IMAGES.

Image sensor model	Pixel pitch [ $\mu\text{m}$ ]		
	Setting 1	Setting 2	Setting 3
ISM1	1.5	2.0	2.5
ISM2	2.3	2.8	3.3

this paper can be applied to publicly available sensor models such as ISET [17]–[20]. Therefore, both models are labeled as image sensor model 1 and 2 (*ISM1* and *ISM2*), shown in Fig. 1. They describe the image sensors with their pixel designs, resolution and other design parameters. In this paper, the values for the pixel pitch are tuned for *ISM1* and *ISM2*. The values are shown in Table I. Since the sensor models are different, the values for the pixel pitch, and Full Well Capacitance (FWC) are different. For *ISM1* the FWC was set to  $15000e^-$  and for *ISM2* the FWC was set to  $10000e^-$ . The FWC values were not changed for tuning the pixel pitch values. The image sensor models generate images with sensor specific artifacts based on their simulated design parameters with resolutions of 500x500, 400x400, 300x300, 200x200 and 100x100 pixels. All generated images were used to create a bit reduced dataset from 8 bits down to 1 bit with normalized values between 0 and 255. In total there are 31200 images, 5200 for each pixel pitch setting of the image sensor models.

#### A. Pupil Detection Algorithm

An existing pupil detection algorithm [21], [22] was modified in [12] and the modified version [12] was taken for the evaluation of the image sensor design parameters. An overview of the algorithm stages is shown in Fig 2. The pupil detection algorithm uses gray scale images as an input. With integral images and Haar-like features a rough estimation of the pupil location is done. For this region estimation a minimum and maximum radius must be selected. In this work, we used 3 different radii settings. The values were normalized for the different resolutions and rounded to integer values, shown in Table II. With the estimation of the pupil region, an intensity-based segmentation with the k-means algorithm is performed. Furthermore, Gaussian filter and morphological

<sup>1</sup><https://www.mathworks.com/products/matlab.html>

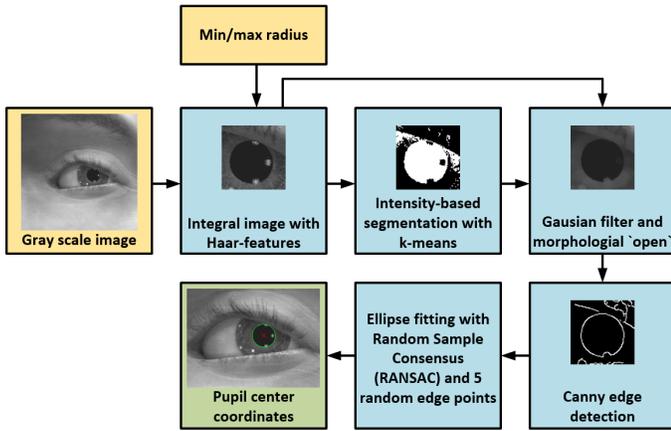


Fig. 2. Overview of the used pupil detection algorithm.

TABLE II  
MINIMUM AND MAXIMUM RADII SETTINGS OF THE ALGORITHM FOR THE EVALUATION WITH DIFFERENT RESOLUTIONS.

Resolution	Radii settings [pixel]		
	Setting 1	Setting 2	Setting 3
500x500	17/38	55/85	17/85
400x400	13/31	44/68	13/68
300x300	10/23	33/51	10/51
200x200	6/16	22/34	6/34
100x100	3/8	11/17	3/17

'open' operations are used to smooth and denoise the image of the estimated pupil region. Then, Canny edge detection is used to find edges. 5 randomly selected edge points are taken for ellipse fitting with Random Sample Consensus (RANSAC). The center coordinates of the best fitted ellipse are taken and compared with the ground truth data. Detailed information of the pupil detection algorithm can be found in [22]. Due to the randomly taken edge points, the pupil center coordinates were calculated 10 times for each image of the whole dataset. The pixel error was calculated based on the processed pupil center coordinates with the algorithm and the ground truth data. The pixel error is the Euclidean distance from the processed pupil center coordinates to the ground truth center coordinates. Additionally, the average pupil detection rate was calculated. The detection rate shows how often the pupil center coordinates were detected for a given pixel error value. Usually, a pixel error up to 5 is considered as a correct detection. For the different resolutions, the pixel errors were normalized to values for a resolution of 500x500 pixels.

### B. Dataset

The dataset consists of images with resolutions of 500x500, 400x400, 300x300, 200x200 and 100x100 pixels with different gaze directions, pupil sizes, eye lid positions, 5 iris colors and ground truth data. The rendered images with 8 bits and a resolution of 500x500 pixels from [12] were used as inputs for the image sensor models to generate the simulated sensor output with resolutions of 500x500, 400x400, 300x300, 200x200 and 100x100 pixels. These images were used to generate bit

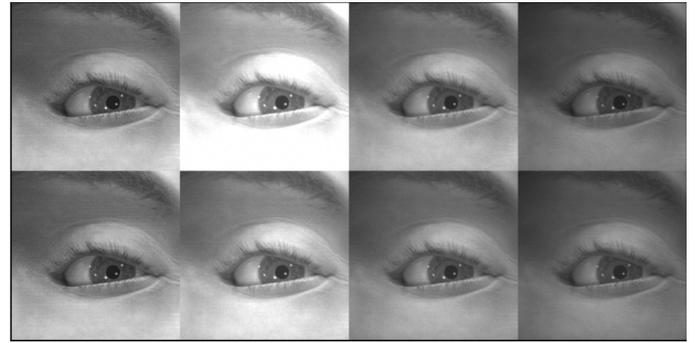


Fig. 3. Example of images from the data set. The top row shows the rendered image on the left and 3 images simulated with ISM1 with pixel pitch setting 1, setting 2 and setting 3 and the bottom row shows also the rendered image on the left and 3 images simulated with ISM2 with pixel pitch setting 1, setting 2 and setting 3.

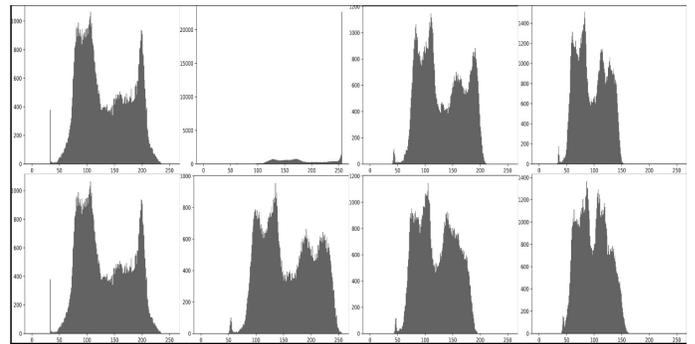


Fig. 4. Comparison of the histograms of the images from Fig. 3. Again, left the rendered images. In the top row 3 images generated with ISM1 and in the bottom row 3 images generated images with ISM2 with the different pixel pitch values.

reduced images with bit depths from 8 bits down to 1 bit. Each image sensor model was configured with 3 different values for the pixel pitch. An example is shown in Fig. 3 with the rendered image (left) and the generated images with pixel pitch values (Table I). The Matlab models scale the illumination power inversely to square of the pixel pitch. This influences the output of the models, which can be observed in the histograms of the generated images. The histograms get shifted and compressed for higher pixel pitch values, shown in Fig. 4. For smaller pixel pitch values, the illumination power increases and the histograms shift towards brighter color values. The ISM1 with a pixel pitch of  $1.5\mu\text{m}$  has a high number of very bright pixels. This has an influence of the pupil detection rate, which will be discussed in the next section. An example of an image from the dataset is shown in Fig. 5 with the different resolutions and different bit depths.

## IV. RESULTS

The whole dataset was processed 10 times with the algorithm and the average pupil detection rate was calculated. Usually, a pixel error up to 5 is considered as a correct detection. For a comparison with the different resolutions, the pixel error was normalized to a resolution of 500x500.

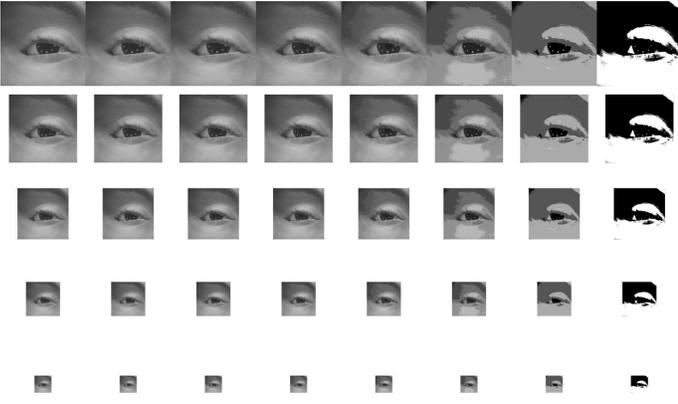


Fig. 5. Example of the dataset with resolutions 500x500, 400x400, 300x300, 200x200 and 100x100 with bit depths from 8 bits down to 1 bit (left to right).

The comparison of the image sensor parameters for a pixel error of 5 is shown in Table III. Detection rates over 25% are highlighted. The results with a resolution of 500x500 pixels are similar compared to 400x400 and therefore not added into the table. However, the performance with a resolution of 100x100 pixels is much lower compared to 300x300 and 400x400 and was not considered for further investigations. The algorithm radii setting 1 (RS1) performs better compared to all other radii settings (RS). However, the average detection rates change with the image sensor models, pixel pitch values, bit depths and resolutions.

For ISM1 the best detection rate is 32.46% with radii setting 1, pixel pitch setting 2, 400x400 resolution and a bit depths of 8 bits. Furthermore, for 300x300 resolution and 4 bits, the average detection rate is only around 2% lower. For a smaller pixel pitch (setting 1), the average detection rate is nearly the same for 4 bits compared to 8 bits with a 400x400 resolution. Also, for 300x300 the detection rate drops around 2%, but is still above 29%. A more detailed look into this configuration shows, that 2 and 3 bits perform better, shown in Fig. 6. Here, the influence of the pixel pitch on the illumination power can be seen for the generated image. The histogram is shifted towards brighter values, which then effects the pupil detection rate. For a resolution of 300x300 and 3 bits, the detection rate is 33.53% and for 2 bits 36.84%, shown in Fig. 6. For a resolution of 400x400 with 3 bits a detection rate of 39.30% is achieved and for 2 bits 39.23%. However, for higher pixel pitch values for ISM1, the detection rate drops with reduced bit depths and resolution. 2 and 3 bits are very sensitive to illumination power and changes in pixel design parameters. Therefore, they are unreliable for image sensors.

For ISM2, again the radii setting 1 gives the best results. Furthermore, a reduction of the bit depth to 4 bits performs better compared to 8 bits for the different resolutions for radii setting 1, shown in Table III. Even for a resolution of 300x300 pixels and 4 bits, the detection rate is above 30%. With pixel pitch setting 2 (PP2) and a resolution of 300x300, the detection rate is even higher, compared to 400x400 with 8 bits. The

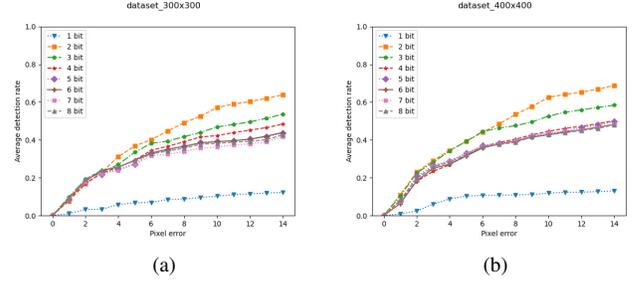


Fig. 6. Average detection rate for ISM1 with pixel pitch of 1.5  $\mu\text{m}$  for resolutions (a) 300x300 and (b) 400x400 pixels and bit depths from 8 bits down to 1 bit.

TABLE III  
RESULTS OF THE AVERAGE PUPIL DETECTION RATE FOR DIFFERENT DESIGN PARAMETER SETTINGS FOR IMAGE RESOLUTIONS OF 200x200, 300x300 AND 400x400 PIXELS.

Settings	Image Parameter					
	200x200		300x300		400x400	
	4 bits	8 bits	4 bits	8 bits	4 bits	8 bits
ISM1_PP1_RS1	<b>25.76</b>	23.84	<b>29.30</b>	<b>28.92</b>	<b>32.07</b>	<b>32.15</b>
ISM1_PP1_RS2	18.00	17.30	16.84	14.61	15.69	15.53
ISM1_PP1_RS3	<b>28.07</b>	<b>26.69</b>	22.84	23.07	<b>25.23</b>	24.46
ISM1_PP2_RS1	<b>27.84</b>	23.46	<b>29.53</b>	<b>26.07</b>	<b>31.15</b>	<b>32.46</b>
ISM1_PP2_RS2	16.15	18.15	16.00	13.53	16.07	16.92
ISM1_PP2_RS3	22.92	23.30	24.76	<b>26.30</b>	23.00	23.46
ISM1_PP3_RS1	23.38	23.30	<b>26.00</b>	<b>29.07</b>	<b>30.53</b>	<b>31.69</b>
ISM1_PP3_RS2	18.92	19.30	16.15	16.07	16.23	18.30
ISM1_PP3_RS3	22.07	23.23	21.92	<b>25.46</b>	22.53	23.15
ISM2_PP1_RS1	<b>25.84</b>	23.23	<b>31.69</b>	<b>28.46</b>	<b>33.69</b>	<b>32.76</b>
ISM2_PP1_RS2	16.53	17.38	16.23	14.30	15.07	15.07
ISM2_PP1_RS3	<b>26.61</b>	23.61	<b>28.15</b>	24.84	24.30	23.69
ISM2_PP2_RS1	24.61	24.15	<b>33.30</b>	<b>29.07</b>	<b>34.92</b>	<b>31.69</b>
ISM2_PP2_RS2	18.23	18.23	15.53	16.00	13.84	18.53
ISM2_PP2_RS3	22.69	24.23	24.15	24.92	24.23	23.46
ISM2_PP3_RS1	<b>28.07</b>	23.00	<b>32.30</b>	<b>27.76</b>	<b>35.38</b>	<b>33.07</b>
ISM2_PP3_RS2	16.07	17.84	17.38	14.84	15.61	17.07
ISM2_PP3_RS3	<b>25.38</b>	23.23	<b>28.00</b>	24.07	<b>28.23</b>	23.69

ISM2 is more suitable for pupil detection with optimizations of bit depths and resolution.

Regardless of the fact that 4 bits for ISM2 performs better compared to 8 bits for certain configurations, a more detailed look into other bit depths is also desired. The average detection rate for ISM2 with different pixel pitch settings, radii setting 1 and resolutions of 200x200, 300,300, 400x400 and 500x500 is shown in Fig. 7 to Fig. 9. The results of ISM2 with pixel pitch setting 1 (PP1) is shown in Fig. 7. The discussion for 1 bit can be neglected, the performance is too low and 1 bit does not work. However, the average detection rates for 2 and 3 bits are higher compared to other bit depths for all resolutions. If the pixel pitch (PP) is changed to setting 2, 2 and 3 bits perform similar to all the other bit depths, except for 1 bit. If the pixel pitch is increased even further to setting 3, 2 and 3 bits perform worse compared to the other bit depths for all resolutions.

The pixel pitch (PP) influences the performance of the algorithm for 2 and 3 bits. This is due to the different illumination of the images and the changed histograms. An example

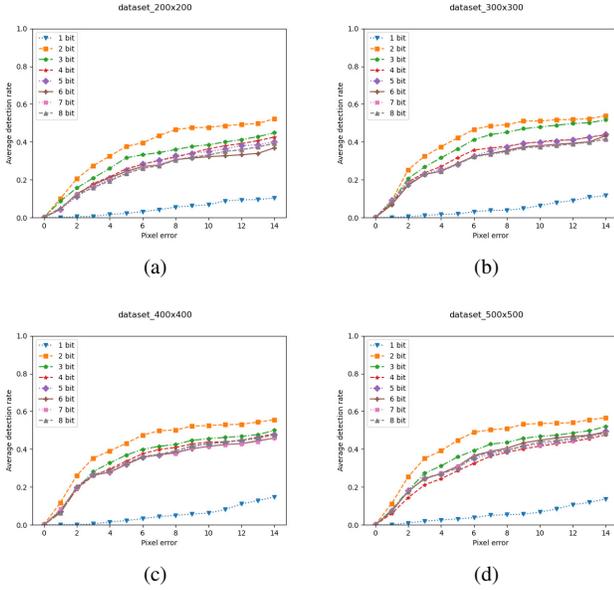


Fig. 7. Average detection rate for ISM2 with pixel pitch of  $2.3 \mu\text{m}$  for resolutions (a)  $200 \times 200$ , (b)  $300 \times 300$ , (c)  $400 \times 400$  and (d)  $500 \times 500$  pixels and bit depths from 8 bits down to 1 bit.

is shown in Fig. 4. The histogram gets shifted for lower pixel pitch values towards brighter colors, since the Matlab model scales the illumination power inversely to square of the pixel pitch. For higher pixel pitch values the histograms get shifted and compressed. Based on this results, optimized image sensors can be designed for specific applications to increase the performance of the image processing system. For pupil detection 4 bits work nearly as good as 8 bits and for ISM2, the performance is even higher. Also,  $300 \times 300$  resolutions work almost as good as  $400 \times 400$  for specific settings. A resolution of  $500 \times 500$  gives similar results as  $400 \times 400$ . A loss of performance becomes visible at a resolution of  $200 \times 200$  pixels. 2 and 3 bits are very sensitive to design parameter changes and are not reliable for future image sensors.

## V. CONCLUSION AND FUTURE WORK

This paper introduces a framework for the optimization of image sensor design parameters, namely pixel design with pixel pitch, image resolution and bit depths for next generation AR/VR image sensors. 2 different image sensor models are used and a pupil detection algorithm is used for the evaluation of the generated images. Furthermore, the image sensor models and pixel pitch values were used to extend an existing pupil detection dataset with different resolutions and bit depths. ISM2 performed better for this pupil detection use case compared to ISM1. The results show, that the image sensor design parameters can be optimized to a bit depth of 4 bits and to a resolution of  $300 \times 300$  pixels. The results for 2 and 3 bits are very sensitive to design parameter changes and not reliable for future AR/VR image sensors.

Future work is to investigate other sensor design parameters and the illumination for further optimizations. Additionally,

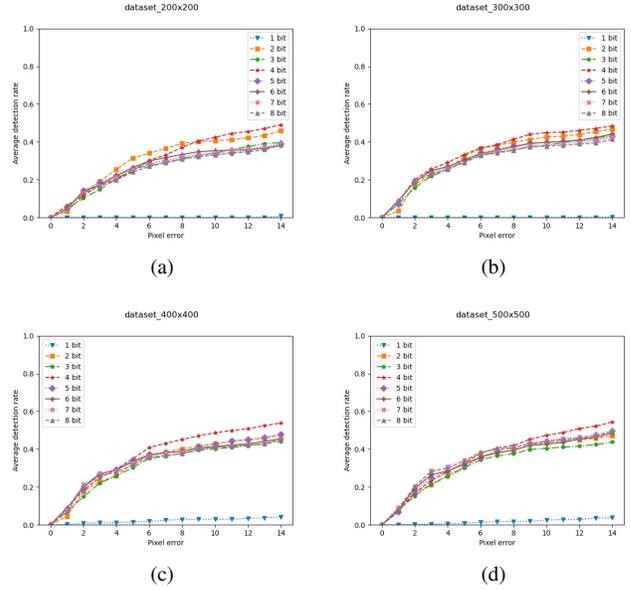


Fig. 8. Average detection rate for ISM2 with pixel pitch of  $2.8 \mu\text{m}$  for resolutions (a)  $200 \times 200$ , (b)  $300 \times 300$ , (c)  $400 \times 400$  and (d)  $500 \times 500$  pixels and bit depths from 8 bits down to 1 bit.

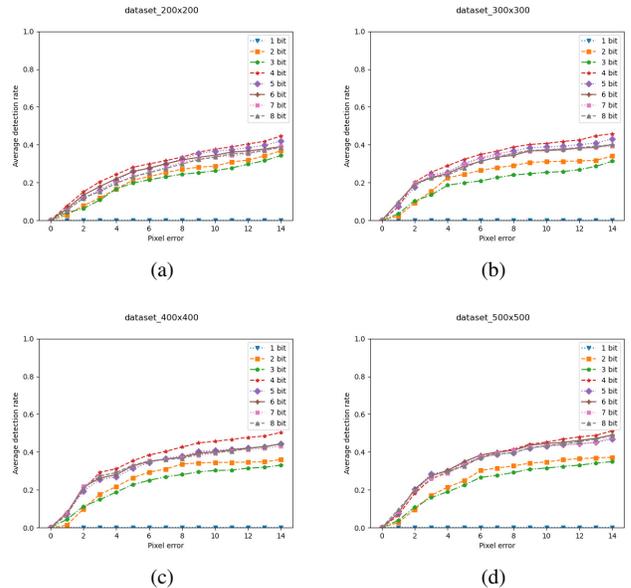


Fig. 9. Average detection rate for ISM2 with pixel pitch of  $3.3 \mu\text{m}$  for resolutions (a)  $200 \times 200$ , (b)  $300 \times 300$ , (c)  $400 \times 400$  and (d)  $500 \times 500$  pixels and bit depths from 8 bits down to 1 bit.

the framework should be extended with additional algorithms, mainly neural network-based models. Furthermore, a bigger dataset should be rendered and generated with the image sensor models to train and test neural networks.

## ACKNOWLEDGMENT

A big thank you goes to my colleagues from TUGraz and ams-OSRAM AG who supported this work with discussions and feedback.

## REFERENCES

- [1] R. Silva, M. Rudek, A. Szejka, J. O. Canciglieri, "Machine Vision Systems for Industrial Quality Control Inspections", 15th IFIP WG 5.1 International Conference, Turin, Italy, isbn: 978-3-030-01613-5, 2018, doi: 10.1007/978-3-030-01614-2\_58
- [2] W. Zhou, M. Fei, H. Zhou, K. Li, "A sparse representation based fast detection method for surface defect detection of bottle caps", *Neurocomputing*, vol. 123, pp. 406–414, 2014.
- [3] P. Horputra, R. Phrajonthong, P. Kaewprapha, "Deep Learning-Based Bottle Caps Inspection in Beverage Manufacturing and Packaging Process", 2021 9th International Electrical Engineering Congress (iEECON), pp. 499–502, 2021, doi: 10.1109/iEECON51072.2021.9440326.
- [4] J. Manzano, T. Bolaños, M. Muñoz, V. Arenas, H. Orozco, "Evaluation of a Machine Vision System Applied to Quality Control in a Liquid Filling, Lid and Labeling Line for Bottles", pp. 140–154, isbn: 978-3-030-62553-5, October 2020, doi: 10.1007/978-3-030-62554-2\_11.
- [5] S.-H. Huang, Y.-C. Pan, "Automated visual inspection in the semiconductor industry: A survey", *Computers in Industry*, vol. 66, pp. 1–10, issn: 0166-3615, 2015, doi: 10.1016/j.compind.2014.10.006.
- [6] M. Tripathi, D. Maktedar, "A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields: A Survey", *Information Processing in Agriculture*, vol. 7, 2019, doi: 10.1016/j.inpa.2019.07.003.
- [7] C. Liu, M. Hall, R. De Nardi, N. Trail, R. Newcombe, "Sensors for Future VR Applications", 2017 International Image Sensor Workshop (IISW), pp. 250–253, 2017.
- [8] C. Liu, A. Berkovich, S. Chen, H. Reyserhove, S.S. Sarwar, T.-H. Tsai, "Intelligent Vision Systems – Bringing Human-Machine Interface to AR/VR", 2019 IEEE International Electron Devices Meeting (IEDM), pp. 10.5.1–10.5.4, 2019, doi: 10.1109/IEDM19573.2019.8993566.
- [9] G. Fiala, J. Loinig, C. Steger, "Impact of image sensor output data on power consumption of the image processing system", *Intelligent Systems Conference 2022. IntelliSys 2022*, 2022, in press.
- [10] S. Paul, K. Rao, G. Coviello, M. Sankaradas, O. Po, Y. Hu, S. Chakradhar, "CamTuner: Reinforcement-Learning based System for Camera Parameter Tuning to enhance Analytics", 2021.
- [11] T. Nürnberg, M. Schambach, D. Uhlig, M. Heizmann, L. León, "A simulation framework for the design and evaluation of computational cameras", in *Proceedings of Automated Visual Inspection and Machine Vision III*, vol. 11061, 2019, 10.1117/12.2527599.
- [12] G. Fiala, Z. Ye, C. Steger, "Pupil Detection for Augmented and Virtual Reality based on Images with Reduced Bit Depths", 2022 IEEE Sensors Applications Symposium (SAS), 2022, in press.
- [13] R. Pinkham, T. Schmidt, A. Berkovich, "Algorithm-Aware Neural Network Based Image Compression for High-Speed Imaging", 2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR), pp. 196–199, IEEE Computer Society, 2020, doi: 10.1109/AIVR50618.2020.00040.
- [14] R. Pinkham, A. Berkovich, Z. Zhang, "Near-Sensor Distributed DNN Processing for Augmented and Virtual Reality", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 4, no. 4, pp. 663–676, 2021, doi: 10.1109/JETCAS.2021.3121259.
- [15] A. Howard, R. Pang, H. Adam, Q. Le, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, "Searching for MobileNetV3", *IEEE/CVF Int. Conf. Comput. Vis.*, pp. 1314–1324, 2019, doi: 10.1109/ICCV.2019.00140.
- [16] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90
- [17] J. E. Farrell, F. Xiao, P. B. Catrysse, B. A. Wandell, "A simulation tool for evaluating digital camera image quality", In *Proc. SPIE* vol. 5294, pp. 124–131, Image Quality and System Performance, Miyake and Rasmussen (Eds.), January 2004.
- [18] J. E. Farrell, P. B. Catrysse, B. A. Wandell, "Digital camera simulation", *Applied Optics* Vol. 51, Iss. 4, pp. A80–A90, 2012.
- [19] J. E. Farrell, B. A. Wandell, "Image Systems Simulation", *Handbook of Digital Imaging* (Edited by Kriss), Chapter 8, isbn: 978-0-470-51059-9, 2015.
- [20] Stanford Center for Image Systems Engineering , "isetcam", <https://github.com/ISET/isetcam>.
- [21] L. Swirski, "Pupiltracker", <https://github.com/LeszekSwirski/pupiltracker>.
- [22] L. Swirski, A. Bulling and N. Dodgson, "Robust real-time pupil tracking in highly off-axis images", In *Proceedings of Eye Tracking Research and Applications Symposium (ETRA)*, pp. 173–176, March 2012, doi: 10.1145/2168556.2168585.