DCNN-Based Screw Detection for Automated Disassembly Processes

1st Erenus Yildiz
III.Physics Institute
Georg-August University of Göttingen
Göttingen, Germany
eyildiz@gwdg.de

2nd Florentin Wörgötter
III.Physics Institute
Georg-August University of Göttingen
Göttingen, Germany
worgott@gwdg.de

***Abstract*—Automation of disassembly processes in electronic waste recycling is progressing but hindered by the lack of automated procedures for screw detection and removal. Here we specifically address the detection problem and implement a universal, generalizable, and extendable screw detector which can be deployed in automated disassembly lines. We selected the best performing state-of-the-art classifiers and compared their performance to that of our architecture, which combines a Hough transform with a novel integrated model of two deep convolutional neural networks for screw detection. We show that our method outperforms currently existing methods, while maintaining the high speed of computation. Data set and code of this study are made public.**

***Index Terms*—screw detection; object detection; automation; autonomous robots; disassembly; recycling**

I. Introduction

Disassembly of electronic devices is a multi-million dollar industry, because of the short life cycle of the current products, which leads to massive amounts of valuable raw material when recycling. The (questionable) trend to increase replacement frequency of electronic devices, such as computers, storage devices, PCBs, etc., this industry is set to grow even further. Electronic products are, thus, usually discarded before their materials degrade. These complex End-of-Life (EOL) products contain a broad spectrum of materials including valuable metals such as silver and copper as well as rare earth metals. When considering only Japan, in their recycling plants currently there are an estimated 300,000 tons of rare earths stored in unused electronics [1]. In France, the Rhodia group is setting up two factories, in La Rochelle and Saint-Fons, that will produce 200 tons of rare earths a year from used fluorescent lamps, magnets and batteries [2]. Therefore, recovery of materials in electronic EOL products is a highly viably business.

Economical reasons are definitely not the only ones for companies and governments investing in increasing the efficiency of recycling processes. Studies like [3] show that recycling processes are either manual or when fully automated the most common way to recycle materials follows the ”crush and separate” paradigm, where the to-be-recycled product is first ground down to small pieces from which the raw materials are then extracted using physico-chemical methods. However, some electronic devices, which arrive to recycling plants, such as GSM amplifier boxes, contain hazardous elements (like Beryllium in the case of GSM amps.). Elements like these represent a massive danger to health and environment. Therefore ”crush and separate” paradigm is not possible in such cases and most of the times human workers have to (pre-)disassemble those with a remaining danger of accidents even in spite of best protective actions. This problem is amplified in countries with less strong health and safety standards. This shows that also health&safety as well as environmental protection render strong incentives for improved automation in recycling.

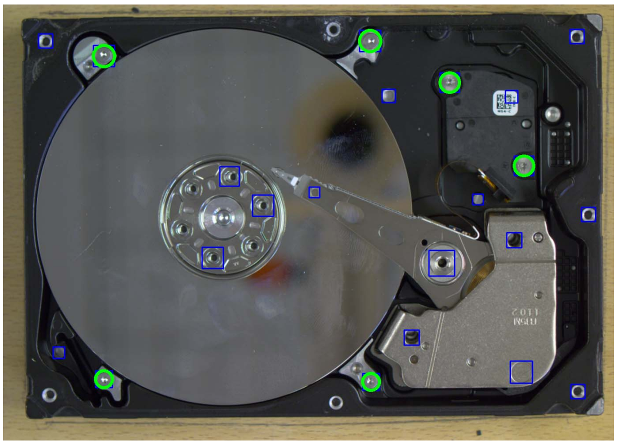

Fig. 1. We present a universal, extendable, RGB based screw detection scheme which is engineered for automated disassembly tasks. Our scheme uses Hough Transform and an integrated model of two deep neural networks. Blue squares indicate possible candidates, while green circles represent the final prediction of screws.

Due to many electronic device variants, there is an urgent need to find a generic solution for automated disassembly, and primarily a generic solution to detect the most essential part of any device when it comes to disassembly: screws. In this paper, a visual screw detection scheme based on the combination of deep learning methods empowered with classical computer vision methods is proposed. The proposed scheme uses the well known Hough transform to generate screw-candidates, which are then filtered by our integrated model, which combines two deep neural networks. The scheme can account for any type

978-1-7281-5686-6/19/$31.00 ©2019 IEEE
DOI 10.1109/SITIS.2019.00040

of screw as long as the user collects enough data for training. Contrary to state-of-the-art techniques, which require users to find specific datasets online, we let users to create their own dataset, given the device. This eliminates the trouble of finding datasets for specific screw types, and ensures high accuracy for our network. Thus, one main approach of this paper focuses on fusion of features acquired from classical computer vision methods and the deep learning model to detect the screws with the dataset, which is collected by the user in a semi-automated way.

We examine the performance of our proposed scheme in different situations and the extent of generalization for effective automation and robotics usage in disassembly. For our experiments, we have collected in total 10000 images of screws as positive samples as well as non-screw artifacts (i.e., holes, stickers, PCB parts, etc.) as negative samples. Data and code are currently being released to facilitate future research.

II. Related Work

Automated disassembly has been researched for a while now [4]–[6] and there are some schemes [7]–[11] for automating certain processes. However, none of the proposed schemes is actually offering a generalizable, extensible, and universal solution of the screw detection problem. Various algorithms have been proposed for screw detection as a part of automated disassembly strategies. Most of these were either too much model-dependent; meaning screw-specific (i.e. only Torx8) or device-specific (i.e., only electric motor screws) [9], [12] or they were extremely brittle due to the fact that the methods they used were highly dependent on classical computer vision methods, which are easily affected by a slight change of illumination [13]. Also, unlike methods like [9], we do not require a depth sensor (i.e., RGB-D camera) to conduct the detection.

Another interesting attempt was conducted by [13]. The authors tried to perform screw detection using template matching on metal ceiling structures for dismantling and successful reuse of light steel gauges. In this scheme, a hierarchical vision system detects the light steel gauge first and then uses multiple template matching to detect screws. This very method also has a very obvious shortcoming: the method depends on a fixed template and therefore it cannot generalize. Also, back then, a light steel gauge had only one type of screw, but there is no guarantee that it will stay like this in the future. Changing the template is tedious and non-desirable and, thus, this method is highly specific and it also cannot address other metal structures or E-Waste devices.

One work, which took our attention, focuses on the disassembly of the electric vehicle batteries using a robot system [6]. Their main goal was to detect M5 bolts on the battery joints. They used a Haar-type cascade classifier, which is trained on cropped images of M5 bolts. Then, to improve the performance of classification, false positives detected from the classifier were added to the negative set. Although the approach sounds quite feasible, unfortunately Haar cascades are not performing very good when it comes to classification. They were able to achieve only 50% detection accuracy, which makes the method impractical for industrial use.

Another work we would like to mention focused on autonomous disassembly of electric vehicle motors [9]. The authors tried to detect screws found on electric vehicle motors using an RGB-D sensor (Kinect) [14]. Although the proposed algorithm is scale, rotation, and translation invariant, it heavily relies on traditional computer vision methods such as Harris corner detection and HSV image analysis, which are easily affected by the lighting conditions. Another shortcoming is the fact that they require a depth image from the RGB-D sensor to remove false positives such as holes, which adds computational load.

Thus, it seems that there is still a substantial lack in generalizable, device and screw-independent methods, which can be used in disassembly processes.

III. Method

In this section we explain each block found in our pipeline. However, before doing that, we would like to inform the reader about the setup our scheme requires. We propose a setup in which the camera faces the device's surface perpendicularly. The distance between the device and the camera was 60 cm, however, depending on the size of the device, this distance may change. Since we worked with computer hard drives, 60 cm was a suitable height.

Our scheme has two modes: offline and online. In the offline mode, the aim is to collect positive and negative samples for the training of the deep neural networks that we use. Therefore, in the offline mode, we are saving possible candidates, which could be screws or artifacts cropped from the camera image. These images are then to be divided by a human into positive and negative samples (screw/non-screw) for the training session. Fig. 2 illustrates the offline mode on the right side.

Having collected the training data and trained the network, the second mode of our scheme is ready to use. When it comes to inferring of screws, as shown in Fig. 2, we again perform the same initial function blocks, however we differ later on and use the trained model of our network to differentiate between positive and negative candidates relying the the trained network. Our scheme then marks and returns the locations of the screws seen in the image.

1) Preprocessing: Computer vision is first used to crop the image to only the region where the device is visible. Cropping is done in a parameter dependent way, so that – depending on the device – users can crop the incoming image as required. Following this, we convert the RGB image into a Grayscale image.

2) Candidate Generation: We have analyzed the different types of screws found in the domain of E-Waste, to make sure that our method will cover all conventionally used screws found in this domain. For this we assessed various electronic devices, which can be found in myriad numbers nowadays in E-Waste, such as computer hard drives, DVD players, gaming consoles and many more. As expected, we found

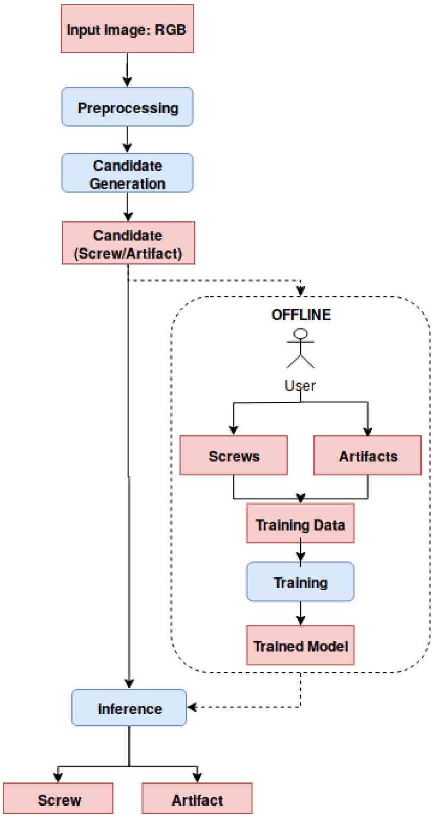

Fig. 2. The pipeline in our scheme is composed of offline and online stages, which are used for training the network and then for inferring screw locations and types using the trained model.

that almost all screws in this domain are circular, which is the natural geometry of a screw and represents the central feature to be used to detect a screw object. Fig. 3 illustrates samples of screws found in E-Waste. There are also non-circular screws manufactured but those are few and we found no such screws in the devices mentioned above. We therefore based our method on first finding circular structures in the images. Obviously, not every circular structure is a screw, for example stickers, holes, transistors, etc. exist, which are also circular, but not screws. Still, circular structures provide us with priors for screws and the first step of our method is to collect those screw-candidates.

As mentioned above, in order to collect candidates, we run our program in offline mode and rely on the *Hough Transform* for candidate detection. This is a standard computer vision method for circle detection [15] and shall not be explained here. Different from the standard Hough Transform here we use a version, which relies on the so called *Hough Gradient* (of the OpenCV library [16]), which uses the gradient information of the edges that form the circle. We refer the reader to the handbook published by the creators of the aforementioned library for further implementation details on the algorithm of the Hough Gradient.

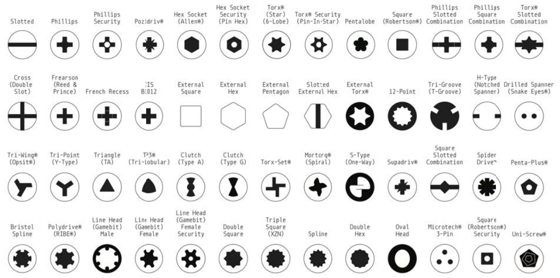

Fig. 3. Screw types encountered during the disassembly of various electronic devices found in E-Waste

3) Training a Classifier: The user manually separates screws from artifacts by which a classifier can be trained using these positive and negative examples as training data. We have investigated also state-of-the-art classifiers found in the literature and we picked the six top-performing ones for comparison at the end. These networks, to our experience, were performing tolerably good given a not so large dataset for a specific device-class (hard drives of any size) class. In Fig. 4 one can see two rows depicting artifacts and screws, respectively, taken from the hard drives. In general, however, these types of positive and negative training samples are observed also in other device-classes and, thus, the resulting training set can be transferred also to other devices. In that case, however, one has to increase the number of samples.

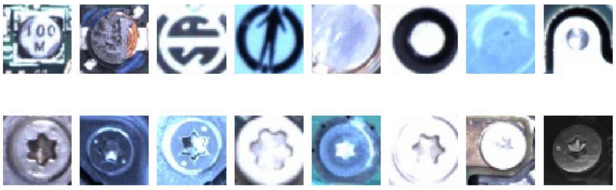

Fig. 4. Candidates collected by the candidate generator in the pipeline. The top row contains artifacts and the bottom row contains screws. The devices analyzed to obtain this were 20 different hard drives of all sizes.

There are several classifiers in the literature, which we could investigate on our task which are Xception [17], InceptionV3 [18], ResneXt101 [19], InceptionResnetV2 [20], Densenet201 [21], Resnet101v2 [22]. These networks achieve over 93% top-5 accuracy on the well-known Imagenet dataset [23]. In order to further improve learning and to reduce overfitting, we inserted a dropout layer before the last fully connected layer of each network.

To further reduce overfitting and to come up with a model that can generalize, we applied an additional data augmentation step. There are several data augmentation operations we applied to introduce more variety in the data. The most important ones are normalizing the image data into a range of [0,1] and randomly setting the brightness in the range [0.5,1.5].

The experimental evaluation each of these network on the test data (see next) allowed us to select two to be used in our processing pipeline.

IV. Experimental Evaluation

We conducted several experiments on the test data we collected. Out of the top six state-of-the-art classifiers, we picked the best performing two and combined them as illustrated in Fig. 5. Below we also provide the details of the experimental evaluation and present our justification for our decision of combining two.

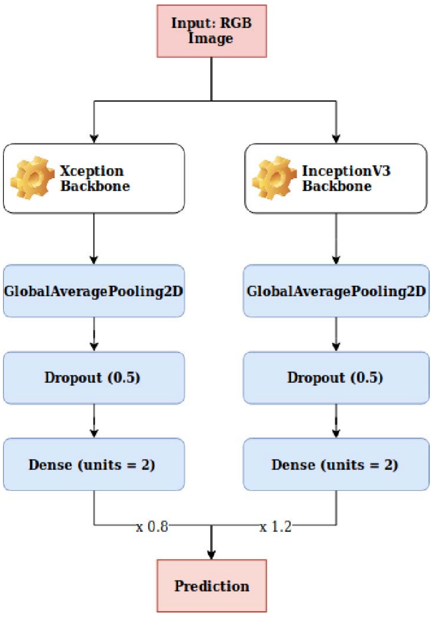

Fig. 5. Our integrated model, which is a combination of the two best performing networks given our data and classification task.

1) Experimental Environment: For the evaluation of the screw classifiers, we collected a dataset consisting of over 10000 samples and split it into training and test sets. The training set includes 1491 screws and 4924 artifacts. The test set includes 1000 screws and 3285 artifacts. We use a computer with Intel Core i7-4770 CPU @ 3.40GHz, 16GB of RAM with GeForce GTX Titan X graphic card to train the classifiers. For evaluation of the performance of our screw detector system, we collected approximately 300 hard drive images containing over 1500 screws. We split those images into training and test sets with ratio of 2:1. To demonstrate the efficiency of our screw detection pipeline, we choose state-of-the-art object detection - YOLOv3 [24], re-train on our screw detection training set and compare results with ours.

2) Experimental Metrics: We use standard metrics for classification evaluation. We are interested in the accuracy of the networks we picked for our pipeline. Therefore we calculate the accuracy of each of them as follows:

$$Acc = (TP + TN)/(TP + TN + FP + FN)$$

where *TP* stands for True Positives, *TN* for True Negatives, *FP* for False Positives and *FN* for False Negatives. To evaluate screw detection performance, we use Average Precision (AP) which is calculated based on the precision-recall curve. For details the reader may refer to [25].

3) Experimental Results: We summarize the experimental results with regards to performance of each classifier against the testing set in Table I.

TABLE I
Experimental results for the chosen state-of-the-art models

Model	TP	TN	FP	FN	Acc
Xception	975	3244	25	41	98.5
InceptionV3	973	3262	27	23	98.8
ResneXt101	962	3260	38	25	98.5
InceptionResnetV2	965	3260	35	25	98.6
DenseNet201	897	3272	103	13	97.3
ResNetV2	943	3208	57	77	96.9
Integrated model	**988**	**3254**	**12**	**31**	**99.0**

From the collected results in Table 1, one can conclude the following: All of the investigated models achieve very high accuracy - over 96% on the testing data, with the model InceptionV3 scoring the highest accuracy of 98.8% among single models. We also notice that the InceptionResnetV2 model scores a very high accuracy of 98.6%, however, note that this model is much heavier than the Xception model which scores comparable accuracy of 98.5%. We then decided to combine two best performing models to boost the accuracy of our classifier. At this point we choose the models InceptionV3 and Xception to build an integrated model for final prediction. Since InceptionV3 performs slightly better than Xception, we prefer to use slightly higher weights on the results of the InceptionV3 model. The final confidence score we use to evaluate the integrated model is presented below:

$$score = (1.2 \times \text{InceptionV3}) + (0.8 \times \text{Xception})$$

Since our scheme uses an integrated model, one has to adjust the value of the confidence level carefully. We have done our experiments to find out the best threshold value for the confidence level, and we chose the threshold for our model by sliding the threshold value over the range from 0.5 to 1.5 with steps of 0.01. Experiments showed that 0.8 is the best threshold and thus is chosen for our integrated model.

We then employ the integrated model in our screw detection pipeline and evaluate it on our test dataset. Fig. 6 shows the precision-recall curve of our screw detector. Our pipeline achieves an AP of 80.23 which clearly outperforms the well known detector YOLOv3 with an AP of 66.47. Fig. 7 illustrates some samples of the detections by our detector.

Discussion and Conclusions

In this study we tackled the fundamental problem of screw detection in disassembly environments. The problem itself is a challenging one, since screws have variable shapes and appearance and not every electronic device has the same

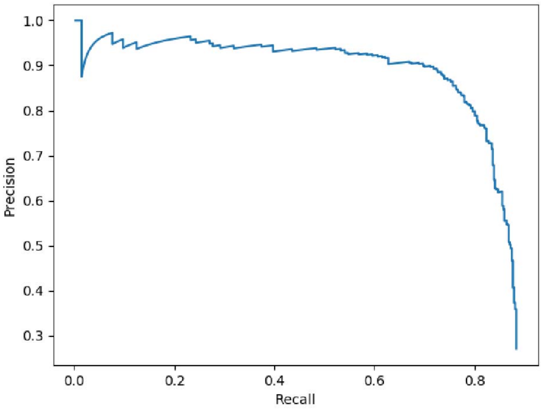

Fig. 6. Precision-Recall Curve

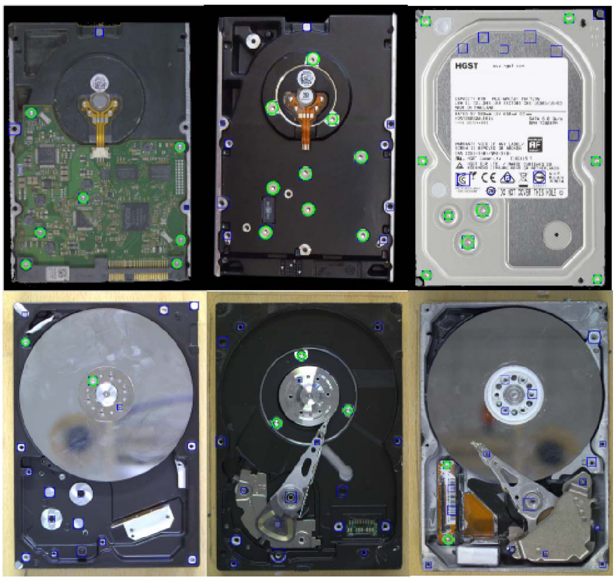

Fig. 7. Sample outputs of the pipeline. Blue rectangles show the candidates generated by Hough Transform and the green circles refer to the true positives predicted by the integrated model. Our scheme is robust to different backgrounds and to different illumination. In all of the images above, the screws were detected correctly.

type of screws. This is the reason why previously developed methods were not useful as a general solution to this problem. We proposed a model, which is based on the Hough transform and deep neural networks. Our scheme easily lets the user use the system for any device of his/her choice, as long as the user separates the collected data into screws and artifacts himself. After doing this and training the network, we could demonstrate that our system achieves real-time performance with quite high accuracy. This had been quantified with hard drive devices of different models and sizes, which have different sizes and types of screws as documented by the experimental evaluation results of our scheme. The data set as well as the ROS-based implementation are published to facilitate further research [1]

Acknowledgments

The research leading to these results has received funding from the European Unions Horizon 2020 Research and Innovation programme (H2020-ICT-2016-1, grant agreement number 731761, IMAGINE; https://imagine-h2020.eu/). We would like to also express our gratitude to Electrocycling GmbH [2] for their continuous support throughout our investigation. Especially we would like to thank Maik Bergamos, who never hesitated sharing his knowledge on recycling as well as the statistical data, along with the devices we requested.

References

[1] H. Tabuchi, "Japan recycles minerals from used electronics," *New York Times*, vol. 4, 2010.

[2] S. Massari and M. Ruberti, "Rare earth elements as critical raw materials: Focus on international markets and future strategies," *Resources Policy*, vol. 38, no. 1, pp. 36–43, 2013. [Online]. Available: https://doi.org/10.1016/j.resourpol.2012.07.001

[3] J. R. Duflou, G. Seliger, S. Kara, Y. Umeda, A. Ometto, and B. Willems, "Efficiency and feasibility of product disassembly: A case-based study," *CIRP Annals*, vol. 57, no. 2, pp. 583–600, 2008. [Online]. Available: https://doi.org/10.1016/j.cirp.2008.09.009

[4] A. Weigl-Seitz, K. Hohm, M. Seitz, and H. Tolle, "On strategies and solutions for automated disassembly of electronic devices," *International Journal of Advanced Manufacturing Technology*, vol. 30, no. 5-6, pp. 561–573, 2006. [Online]. Available: http://dx.doi.org/10.1007/s00170-005-0043-8

[5] K. Dröder, A. Raatz, C. Herrmann, K. Wegener, and S. Andrew, "Disassembly of Electric Vehicle Batteries Using the Example of the Audi Q5 Hybrid System," *Procedia CIRP*, vol. 23, pp. 155–160, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.procir.2014.10.098

[6] K. Wegener, W. H. Chen, F. Dietrich, K. Dröder, and S. Kara, "Robot assisted disassembly for the recycling of electric vehicle batteries," *Procedia CIRP*, vol. 29, pp. 716–721, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.procir.2015.02.051

[7] A. Elsayed, E. Kongar, S. M. Gupta, and T. Sobh, "A robotic-driven disassembly sequence generator for end-of-life electronic products," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 68, no. 1, pp. 43–52, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10846-012-9667-8

[8] J. Pomares, S. T. Puente, F. Torres, F. A. Candelas, and P. Gil, "Virtual disassembly of products based on geometric models," *Computers in Industry*, vol. 55, no. 1, pp. 1–14, 2004. [Online]. Available: http://dx.doi.org/10.1016/j.compind.2004.03.001

[9] M. Bdiwi, A. Rashid, and M. Putz, "Autonomous disassembly of electric vehicle motors based on robot cognition," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June, no. July, pp. 2500–2505, 2016. [Online]. Available: https://doi.org/10.1109/ICRA.2016.7487404

[10] U. Büker, S. Drüe, N. Götze, G. Hartmann, B. Kalkreuter, R. Stemmer, and R. Trapp, "Vision-based control of an autonomous disassembly station," *Robotics and Autonomous Systems*, vol. 35, no. 3-4, pp. 179–189, 2001. [Online]. Available: https://doi.org/10.1016/S0921-8890(01)00121-X

[11] S. Q. Xie, D. Cheng, S. Wong, and E. Haemmerle, "Three-dimensional object recognition system for enhancing the intelligence of a KUKA robot," *International Journal of Advanced Manufacturing Technology*, vol. 38, no. 7-8, pp. 822–839, 2008. [Online]. Available: http://dx.doi.org/10.1007/s00170-007-1112-y

[12] S. R. Cruz-Ramirez, Y. Mae, T. Takubo, and T. Arai, "Detection of screws on metal-ceiling structures for dismantling tasks in buildings," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 4123–4129. [Online]. Available: https://doi.org/10.1109/IROS.2008.4650975

[1] https://github.com/eyildiz-ugoe/screw_detection

[2] http://www.electrocycling.de

[13] H. Ukida, “Visual defect inspection of rotating screw heads,” in *SICE Annual Conference 2007*. IEEE, 2007, pp. 1478–1483. [Online]. Available: https://doi.org/10.1109/SICE.2007.4421216

[14] Z. Zhang, “Microsoft kinect sensor and its effect,” *IEEE multimedia*, vol. 19, no. 2, pp. 4–10, 2012. [Online]. Available: https://doi.org/10.1109/MMUL.2012.24

[15] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines and curves in pictures,” SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER, Tech. Rep., 1971.

[16] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. ” O’Reilly Media, Inc.”, 2008.

[17] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2017.195

[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826. [Online]. Available: https://doi.org/10.1109/CVPR.2016.308

[19] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500. [Online]. Available: https://doi.org/10.1109/CVPR.2017.634

[20] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[21] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708. [Online]. Available: https://doi.org/10.1109/CVPR.2017.243

[22] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in *European conference on computer vision*. Springer, 2016, pp. 630–645. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-46493-0_38

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255. [Online]. Available: https://doi.org/10.1109/CVPR.2009.5206848

[24] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[25] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun 2010. [Online]. Available: https://doi.org/10.1007/s11263-009-0275-4