

Simulation-based Data Augmentation for the Quality Inspection of Structural Adhesive with Deep Learning

Ricardo Silva Peres, *Member, IEEE*, Magno Guedes, Fábio Miranda and Jose Barata *Member, IEEE*

Abstract—The advent of Industry 4.0 has shown the tremendous transformative potential of combining artificial intelligence, cyber-physical systems and Internet of Things concepts in industrial settings. Despite this, data availability is still a major roadblock for the successful adoption of data-driven solutions, particularly concerning deep learning approaches in manufacturing. Specifically in the quality control domain, annotated defect data can often be costly, time-consuming and inefficient to obtain, potentially compromising the viability of deep learning approaches due to data scarcity. In this context, we propose a novel method for generating annotated synthetic training data for automated quality inspections of structural adhesive applications, validated in an industrial cell for automotive parts. Our approach greatly reduces the cost of training deep learning models for this task, while simultaneously improving their performance in a scarce manufacturing data context with imbalanced training sets by 3.1% (mAP@0.50). Additional results can be seen at <https://git.io/Jtc4b>.

Index Terms—Quality Inspection, Deep Learning, Synthetic Data, Simulation, Structural Adhesive.

I. INTRODUCTION

In several sectors of the manufacturing industry, including automotive, naval and aerospace, guaranteeing the safety of the product's end-user is a top priority. This makes it critical to ensure that each manufactured part adheres to strict quality criteria. Typically, a crucial step of quality assurance in these sectors consists in the tests performed after final assembly. However, detecting problems and manufacturing defects only at such a late stage in the process is not desirable for manufacturers, as fixing them becomes either too time consuming and costly, or even impossible resulting in total loss.

To mitigate this, manufacturers integrate quality inspection of parts and components along the production line. These tests are often destructive, being performed by sampling as part of statistical quality control. A particular example of such tests are those carried out for parts bonded with structural adhesive in the automotive industry.

With the increasing demand for lighter and more resistant materials, manufacturers are frequently opting for lighter metal alloys or carbon fibre parts which often can not be welded.

R.S. Peres and J.Barata are with UNINOVA - Centre of Technology and Systems (CTS) and NOVA School of Science and Technology, FCT Campus, Monte de Caparica, 2829-516, Caparica, Portugal

M.Guedes and F.Miranda are with Introsys S.A., Estrada dos 4 Castelos 67, Quinta do Anjo, 2950-805, Portugal

This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

Even in cases where welding is possible, the application of structural adhesive before bonding plays an important role as it contributes to the reduction of noise, vibrations and infiltrations.

In general, destructive quality tests consist in separating bonded parts to analyse the spread, consistency and continuity of the adhesive. This procedure is not only time consuming, but also costly in terms of human resources, materials and waste. Furthermore, common defects such as discontinuities and blobs are impossible to correct if they are not detected before bonding, due to the parts having already undergone mechanical and structural changes at that stage. Discontinuities for instance are critical since they are generally undetectable after bonding. An example of a process for which the rigorous application of structural adhesive is crucial is hem flange bonding [1], illustrated in Figure 1.

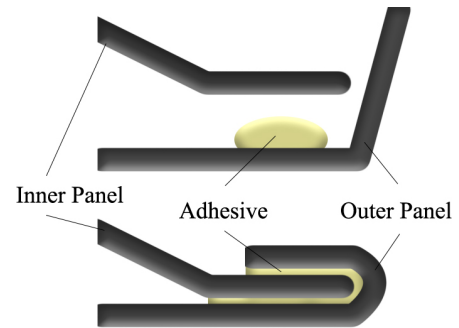


Fig. 1. The hem flange bonding process

This process of metal bonding broadly consists in bending an outer closure panel over an inner panel after applying adhesive between them, being frequently employed in the assembly of doors, liftgates and hoods. If there is a shortage of material, the result will be poorly isolated which can lead to an early corrosion of the part. Contrastingly, if too much adhesive is dispensed it can leak outside the flange and must be cleaned.

Based on these points, enabling the continuous and systematic quality inspection of all parts, beyond traditional statistical quality control, becomes highly desirable for manufacturers striving for zero-defect manufacturing. In this light, recent advances in Artificial Intelligence (AI), particularly concerning deep learning and computer vision are extremely promising in regard to the automation of such inspections [2].

However, deep learning approaches generally require very

high volumes of data. In the particular case of manufacturing quality control, such approaches need considerable amounts of defect data which tends to be costly to generate and thus scarce.

To address this challenge, we design, implement and validate an approach to improve the performance of quality inspection with deep learning using synthetically generated data. Our contributions can be summarized as follows:

- 1) A novel approach to augment image datasets of structural adhesive application processes;
- 2) An open-source simulation to generate synthetic images of structural adhesive beads;
- 3) A publicly available dataset of structural adhesive applications, SEE-Q, consisting in 124 manually annotated images of real adhesive beads. This dataset can be augmented with the provided simulation to reproduce the results described herein;
- 4) Weights and configurations for models trained and validated in a real production cell to support the reproducibility of the results;

The remainder of this article is structured as follows. Section II overviews related work from current literature. Section III addresses the approach for the generation of synthetic data and for training the object detection models, as well as the specification of the scenarios for validation. Then, Section IV describes the validation environment and the corresponding datasets, followed by a discussion of the results. Finally, Section V discusses the limitations of the approach, followed by the conclusions and future work in Section VI.

II. RELATED WORK

Concerning the domain of quality control, most mature organizations have adopted quality-oriented strategies and paradigms, such as lean production and six sigma, in an attempt to minimize defects and scrap. Due to this, in some processes the occurrence of defects can be seen as a relatively rare event [3] (albeit costly), making data availability a major challenge for modern Machine Learning (ML) based approaches.

Moreover, a recent survey¹ found 96% of the respondents across twenty different industries reported training data quality and labelling challenges in ML projects. In the case of manufacturing, examples of factors that contribute to this include the difficulty of collecting exhausting real-world samples, leading to selection bias (e.g., poor performance detecting defects in products in different poses when trained on data containing samples in a single pose), or edge cases which can originate from rare events or simply be too costly to recreate.

As a result, one of the main limitations of modern deep learning-based approaches to quality control are the vast amounts of training data required to develop such solutions, which require considerable human effort, are costly, time consuming and error-prone. In this light, the usage of synthetic data is emerging as an attractive solution to decrease the burden of data collection and annotation [4].

A recent systematic review [5] of Industrial AI applications showcased that roughly 20% of the publications included in the study employed synthetic data to augment their datasets. The generation of synthetic data which closely resembles data from real operational environments is a key driver behind the generalization capability of Industrial AI solutions to real scenarios.

With the advent of Generative Adversarial Networks (GAN) [6], new opportunities for the generation of reliable synthetic data in manufacturing have arisen. The GANs can be used to learn the distributions of the original data and the generate fake, yet realistic samples to expand the training dataset. Some promising examples of this can be found in the literature, addressing applications in which the data of faulty operational conditions is scarce and difficult to obtain [7]–[9]. Recent efforts have also shown promising empirical results on semi-supervised learning [10]–[12] for cases in which labelling the entire dataset is unfeasible or too costly.

However, a large amount of real training data is still necessary in order to train GAN models to generate training data with sufficient quality. While considerable progress has been made towards solving this challenge, a state-of-the-art approach for small datasets developed by researchers at NVIDIA [13] still requires over 1000 images to train. An alternative approach is thus centred on the creation of simulation models capable of closely replicating the manufacturing process.

Recent research from OpenAI has shown that techniques such as domain randomization [14], [15] can be employed to generate synthetic data using simulation in increasingly difficult environments, which in turn can be used to train robust ML models capable of generalizing well to real-world tasks.

The main drawbacks of simulation or 3D rendering-based approaches in current literature are that most are still dependent on real data, or require considerable modelling effort to create photorealistic 3D assets and scene compositions [16], [17]. This can make such approaches unsuitable for the domain of manufacturing quality control, as it is often unfeasible to exhaustively collect sufficient and representative defect data.

In Section III we propose and describe a simulation-based approach through which samples of defective parts can be generated with considerable variation for the quality inspection of structural adhesive applications, whilst requiring minimal modelling effort. Additionally, we showcase that synthetic data generated by this simulation can be used to improve the detection of defects in a real industrial cell, when compared to an off-the-shelf deep learning model trained only on a small real dataset.

III. MATERIALS AND METHODS

This section details the approach to generate synthetic and automatically annotated images to train deep learning object detectors for inline quality inspection. The implementation of the simulation, as well as model configurations, weights and data are available at <https://github.com/RicardoSPeres/simulation-synth-adhesive>.

¹<https://www.businesswire.com/news/home/20190523005183/en/Survey-96-Enterprises-Encounter-Training-Data-Quality>

A. Synthetic Dataset Generation

The simulation environment was implemented in CoppeliaSim [18], consisting in a bare bones scene aimed at visually replicating the end result of the manufacturing process, in this instance the application of industrial structural adhesive to an automotive part. Both the simulated environment and an example of the process outcome can be seen in Figure 2.

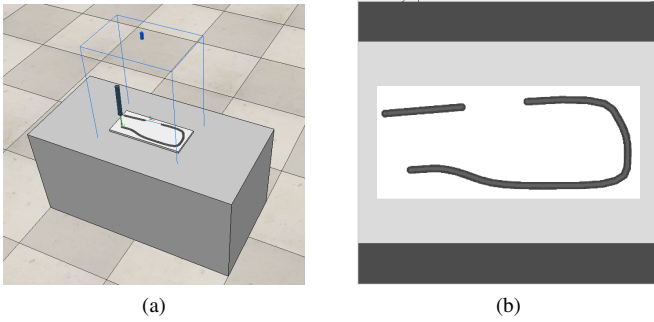


Fig. 2. Simulation environment in CoppeliaSim (a) and an example of a synthetic image generated with a discontinuity defect in the structural adhesive bead (b)

Defects can be generated all along the pre-defined path for the adhesive bead, being constrained to a maximum of three defects per part. These defects can be of two types, one being a discontinuity as shown in Figure 2, the other being characterized by an excess of material (e.g., adhesive blob). However, the simulation can be easily parameterized to generate additional defects or types, such as the narrowing of the bead.

Upon starting the process, an algorithm decides how many defects will be generated and of which type. Then, their position along the bead and length is computed, followed by the actual application of the adhesive. The algorithm is described in Listing 1.

Afterwards, the annotations of the bounding boxes for each defect instance are generated following the YOLO format:

$$\langle object\ class \rangle \langle x \rangle \langle y \rangle \langle width \rangle \langle height \rangle \quad (1)$$

where the *object class* refers to the defect type (i.e., 0 for discontinuity, 1 for excess), x and y are the coordinates for the center of the bounding box and *width* and *height* define its dimensions. The algorithm to generate the annotation file for each image is provided in Listing 2.

The equation that approximates the x coordinate as a function of the defect starting point relative to the path solves as the following 6th order polynomial:

$$x = -6.2428 \times 10^{-9}x^6 + 1.8952 \times 10^{-9}x^5 - 0.0002x^4 + 0.0097x^3 - 0.1929x^2 + 3.1184x + 7.5402 \quad (2)$$

Similarly, the same can process can be applied to the y coordinate of the defect's starting point:

$$y = 3.8936 \times 10^{-9}x^7 - 1.3498 \times 10^{-7}x^6 + 1.8193 \times 10^{-5}x^5 - 0.0012x^4 + 0.0384x^3 - 0.5653x^2 + 3.1013x + 64.2042 \quad (3)$$

Algorithm 1: Defect generation

Result: Number and type of defects with their position relative to the length of the bead

```

defectTypeList = {};
dLengthList = {};
numberOfDefects = math.random(1, 3);
i = 0;
while i < numberOfDefects do
    | defectTypeList[i] = math.random(0,1);
end
for i=0; i < numberOfDefects; i++ do
    if defectTypeList[i]==0 then
        | dLengthList[i] = Random medium length;
    else
        | dLengthList[i] = Random shorter length;
    end
    Compute the defect's start position in the bead as a
    function of numberOfDefects, its length and
    the start position of the previous defect (if it
    exists);
end

```

Algorithm 2: Annotation generation

Result: File containing the annotations for the defects in a single synthetic image

```

foreach defect do
    defectEnd = defectStart + defectLength;
    x_start = convertPathWidthToX(defectStart);
    x_end = convertPathWidthToX(defectEnd);
    y_start = convertPathWidthToY(defectStart);
    y_end = convertPathWidthToY(defectEnd);
    x_center = (x_start + x_end) / 2;
    y_center = (y_start + y_end) / 2;
    width = math.abs(x_end - x_start);
    height = math.abs(y_end - y_start);
    Write the resulting annotation to file following the
    YOLOv4 format
end

```

Based on this, an annotated synthetic dataset of 4000 defect images was generated. A few examples of real defects and their similar synthetic counterparts is showcased in Figure 3.

B. Training Object Detection Models

For the implementation of object detection to automate quality inspection, different model architectures were trained using Darknet [19], an open source neural network framework written in C and CUDA supporting CPU and GPU computation. The base implementation is available at <https://github.com/AlexeyAB/darknet>.

We employed YOLOv4 and Yolov4-Tiny architectures [20], as well as Scaled-YOLOv4 [21] which achieved state-of-the-art results for the MS COCO dataset [22]. The training of these models was carried out for two different scenarios as detailed in Subsections III-B1 and III-B2, using a single

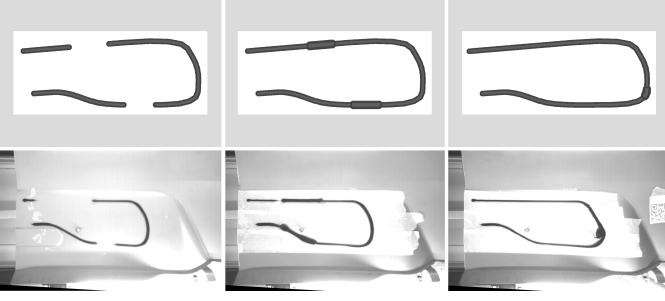


Fig. 3. Examples from the generated synthetic dataset and its real counterparts. First column presents discontinuity defects, the remaining two showcase varied excess defects

NVIDIA GeForce GTX 1660 Ti GPU (Turing architecture, 6 GB GDDR6). For both scenarios, the performance of the models was assessed based on the Mean Average Precision (mAP) at different Intersection Over Union (IoU) thresholds, a common metric in most modern object detection tasks [22], [23]. The overview of the approach is illustrated in Figure 4.

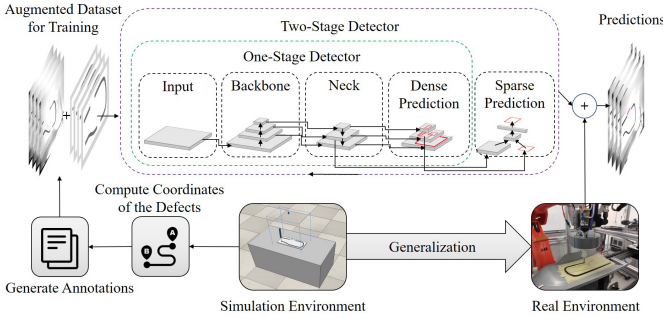


Fig. 4. Overview of the proposed approach. The general object detector is adapted from [20]

1) *Scenario A - Generalizing from Synthetic to Real*: This first scenario was designed to test the capacity of the synthetic models to generalize to the real test set. In this case different state-of-the-art object detection models were trained only on the synthetic dataset (4000 images) and then tested on 124 real images. Transfer learning was used for the initial weights pre-trained on the MS COCO dataset.

2) *Scenario B - Assessing Data Augmentation in Scarce Settings*: The goal of the second scenario was to assess the impact of augmenting the scarce dataset of real defects with the synthetic images generated in simulation. For this purpose an ablation study was carried out for all model variants by removing different parts of the training set (either training on synthetic, real or augmented data) and comparing their performance on a smaller test set of 36 real images (given the limited availability of real images) to assess the impact of the different training sets. Model parameters and configurations such as batch size and learning rate were kept uniform across all tests.

To further assess the impact of the augmentation, different volumes of augmented data were considered. In one experiment, all 4000 synthetic images were added to the real training

set, while on another only 50 synthetic images containing mostly instances of the under-represented class (excess defect) were included.

IV. EXPERIMENTAL RESULTS

In this section the real validation environment is described, along with a thorough discussion of the results obtained in the experiments.

A. Validation Environment

The validation was carried out in a high throughput pilot cell involved in the application of structural adhesive beads for the automotive industry [24]. The demonstrator is located at Introsys S.A. facilities in Castelo Branco, Portugal, a company specializing in industrial automation (particularly in the automotive sector) that operates in the international market since 2004.

Within the context of this study, two complimentary stations are considered, one for the application and correction of the structural adhesive bead with in-process quality inspection, the other for post-process quality inspection. The first encompasses an ABB IRB 2400 6-DoF industrial robot, a Nordson VersaPail bulk melter, a Nordson AG-930s dispensing gun, an in-process inspection ring with 3 IDS OEM cameras and built-in IR led lighting. The post-process inspection setup is composed by two Teledine cameras and IR lighting. The stations are linked by a conveyor system and controlled by a Siemens Simatic S7-400 PLC.

The conveyor system is able to transport up to five pallets. After a part has been loaded into the system, it is transported to the adhesive application station. Upon arrival, the pallet is elevated, the robot applies the structural adhesive bead, then the lift descends and the part is transported to the post-process inspection station, where is inspected using the deep learning model. If the part passes the quality check, or the detected defects are unrecoverable, it is transported to the unloading station and unloaded by a worker. If a recoverable defect is detected (i.e., discontinuity), the part loops back to the first station where the bead can be corrected.

B. The SEE-Q Dataset

The SEE-Q dataset comprises the 124 manually annotated images of real industrial structural adhesive defects used as the basis for this approach. It consists of 160 instances of discontinuity and 70 of excess defects. For reproduction purposes it is made publicly available at <https://github.com/RicardoSPeres/simulation-synth-adhesive>.

For testing the proposed approach, the SEE-Q dataset was augmented with 4000 synthetic images annotated automatically in the simulation environment. An example of the resulting annotated images can be seen in Figure 6.

C. Discussion of Results

Starting with Scenario A, a summary of the results in terms of mAP for various IoU thresholds, as well as of the average detection time per image is provided in Table I.

TABLE I

EVALUATION METRICS FOR THE DIFFERENT MODELS TRAINED WITH 4000 SYNTHETIC IMAGES OVER 6000 ITERATIONS, TESTED ON 124 REAL IMAGES. THE mAP IS COMPUTED FOR DIFFERENT IOU THRESHOLDS, NAMELY 0.15, 0.30 AND 0.50. BEST RESULTS ARE HIGHLIGHTED IN BOLD.

Model	Size	mAP@0.15	mAP@0.30	mAP@0.50	Avg. Detection Time (ms)
YOLOv4x-Mish	640	0.4574	0.4014	0.154	72.58
YOLOv4	512	0.725	0.6392	0.4449	40.32
YOLOv4-Tiny	416	0.5188	0.4684	0.2762	16.13

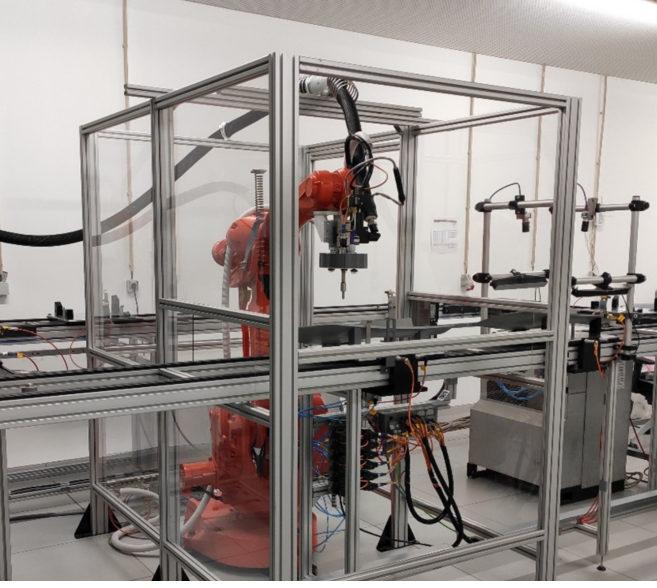


Fig. 5. Overview of the structural adhesive application cell used for validation.

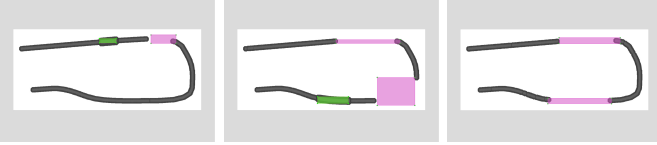


Fig. 6. Example of the synthetic images included in the A-SEE-Q dataset, with automatically generated annotations.

While the mAP is relatively low, particularly when later compared with the results achieved in Scenario B, it serves to show that the models are still capable of generalizing from the simulation to the real environment, being able to detect both types of defects in most cases where the bead did not differ greatly from those observed in the synthetic set. In edge cases, the models can generally still detect the presence of defects, but struggle with the detection of all instances in the input image. An example of these two cases is provided in Figure 7. Unfortunately, the YOLOv4x-Mish model did not achieve comparable performance within the same number of iterations with the experiment's setup.

It is also worth noting that there's a small difference between the background of these images. This stems from the fact that for some samples included in the dataset, a sheet of paper was glued to the product part, with the structural adhesive being applied on top of it. This was done due to

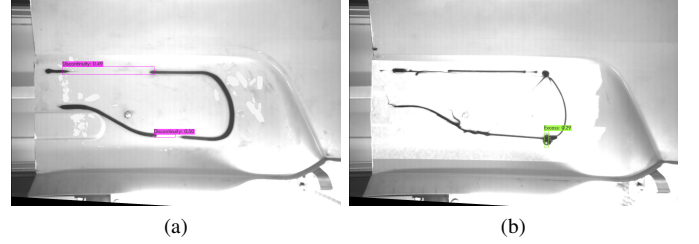


Fig. 7. Example of detections using the YOLOv4 model trained only on synthetic data, with (a) resembling the examples from simulation and (b) representing an edge case.

the inherent difficulty of removing the adhesive from the part, even with the use of solvents, so that parts could be reused for testing. Regardless, the influence of this difference was not noticeable in the results, as the models were robust even in the face of more pronounced changes in the image.

Moreover, this further illustrates the problems concerning the generation of real defect datasets in this context. Firstly, the occurrence of defects is not desirable in normal production, making such cases naturally scarce. Adding to this, even in pilot lines or laboratory environments, the process is extremely time consuming, costly and complex, since constant changes to the equipment's control parameters are required to inject different defects.

Concerning Scenario B, the YOLOv4 and YOLOv4-tiny models were trained on four different training sets, being evaluated on the same test set of real images as described in Table II. As it can be observed, training the YOLOv4 model with the augmented dataset yielded improved performance for mAP at all of the different thresholds measured, consisting in the best results across all training sets. Comparing with the best performing YOLOv4-Tiny model, it can be seen that while YOLOv4 is superior in mAP (around a 9.16% difference in mAP@0.50), the tiny model is nearly twice as fast concerning inference time.

It is also interesting to note that the best tiny model resulted from training on an augmented dataset, albeit with fewer instances mostly of the under-represented class, as opposed to the full 4000 images of the balanced synthetic set.

A comparison of the outputs from the YOLOv4 models trained on synthetic, real and augmented sets is provided in Figure 8. Additional examples of the results can be found at <https://git.io/Jtc4b>.

The rows of images in Figure 8 depict increasingly difficult examples (more and more different from typical defects in the training set). Taking an off-the-shelf model such as YOLOv4 and training it on a small real-world dataset can be considered

TABLE II

RESULTS OBTAINED FOR SCENARIO B, GROUPED BY THE VARIED TRAINING SETS. ALL GROUPS WERE TESTED ON A HOLDOUT SET OF 36 REAL IMAGES CONTAINING 69 INSTANCES OF DIFFERENT DEFECTS.

Model	Size	mAP@0.15	mAP@0.30	mAP@0.50	Avg. Detection Time (ms)
Synthetic Training Set (4000 images)					
YOLOv4 (S)	512	0.6427	0.6108	0.3995	55.56
YOLOv4-Tiny (S)	416	0.4666	0.4497	0.2158	27.78
Real Training Set (88 images)					
YOLOv4 (R)	512	0.9131	0.9016	0.8865	55.56
YOLOv4-Tiny (R)	416	0.8666	0.8526	0.8048	27.78
Augmented Training Set (88 Real / 4000 Balanced Synthetic Images)					
YOLOv4 (A)	512	0.9359	0.9334	0.9173	55.56
YOLOv4-Tiny (A)	416	0.9051	0.8405	0.5652	27.78
Augmented Training Set (88 Real / 50 Synthetic images mostly of the under-represented class)					
YOLOv4 (A*)	512	0.9238	0.9227	0.9006	55.56
YOLOv4-Tiny (A*)	416	0.887	0.8743	0.8257	27.78

common practice, and in fact in this case the model performs well for images with common defects (first and second row). However, performance worsens considerably when faced with edge cases (third row). In general, it was observed that the model trained on the augmented dataset presented tighter bounding boxes during inference with better localization, in some cases even detecting instances missed (false-negatives) by the model trained only on real data.

Finally, an additional experiment was carried out to further test the generalization of the models trained on the augmented data. For this purpose, different paths for the application of the structural adhesive were modelled in the simulation, generating beads with considerably different shapes when compared to those present in the training data. The best YOLOv4 model from Scenario B was then tested on these inputs, with the results shown in Figure 9.

Once more, as it can be verified the model is able to successfully identify both types of defects even in beads with shapes considerably different than the one seen in the training phase. This suggests potential concerning the applicability of the model to applications with varying shapes without requiring additional re-engineering effort.

V. LIMITATIONS

While the proposed approach is robust to variations regarding for instance position or lighting conditions in the quality inspection images, the performance of the models is limited for extreme cases that differ too much from the training set. An example of this can be found in the last row of Figure 8.

Despite this, it can be observed that the model is still capable of detecting the presence of defects, even if in few cases it might be unable to localize all instances in the product. In this regard, one venue that can be explored is to improve the simulation to include a wider variety of such cases.

Additionally, one clear point for improvement is the generation of the annotations, as the localization of the resulting bounding boxes (specifically the approximation of the x and y coordinates) still presents inaccuracies that should be resolved

in future iterations. This is hypothesized to be the reason behind the difference in performance for the tiny model between the two augmented sets presented in Table II, and suggests that improvements over training exclusively with the real set could in reality be larger once this aspect is resolved.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we show that state-of-the-art object detection models are capable of generalizing the learning from purely synthetic data to real scenarios. This holds true even in cases such as the one presented, where the simulation outcome is not a perfect realistic representation of the real environment.

Furthermore, we show that the synthetic data can be used to augment the original small real dataset, resulting in an improved performance for the object detection task. As such, this approach is suitable for instance in scenarios for which there is not enough data to train GAN-based models (which could otherwise be an alternative approach), while requiring minimal modelling effort for the simulation.

Additionally, the proposed approach greatly reduces the costs of generating additional training data with sufficient samples of each defect. Typically, even in controlled environments, the process of generating real images would result in energy, material and personnel costs. For real production environments, it is generally unfeasible to dedicate a line solely for the purpose of generating specific data. For both cases, the proposed approach represents a valid alternative at a very small fraction of the cost.

As future work (beyond addressing the limitations discussed in Section V), if the problems of training GANs with very small datasets are solved (e.g., stabilizing the discriminator with less 500 samples), there is considerable potential in exploring that venue. Generative approaches remove the need to model the simulation for each different use case, while likely providing more life-like results which could further improve the performance obtained from synthetically augmenting the dataset. At that point, the choice for an adequate

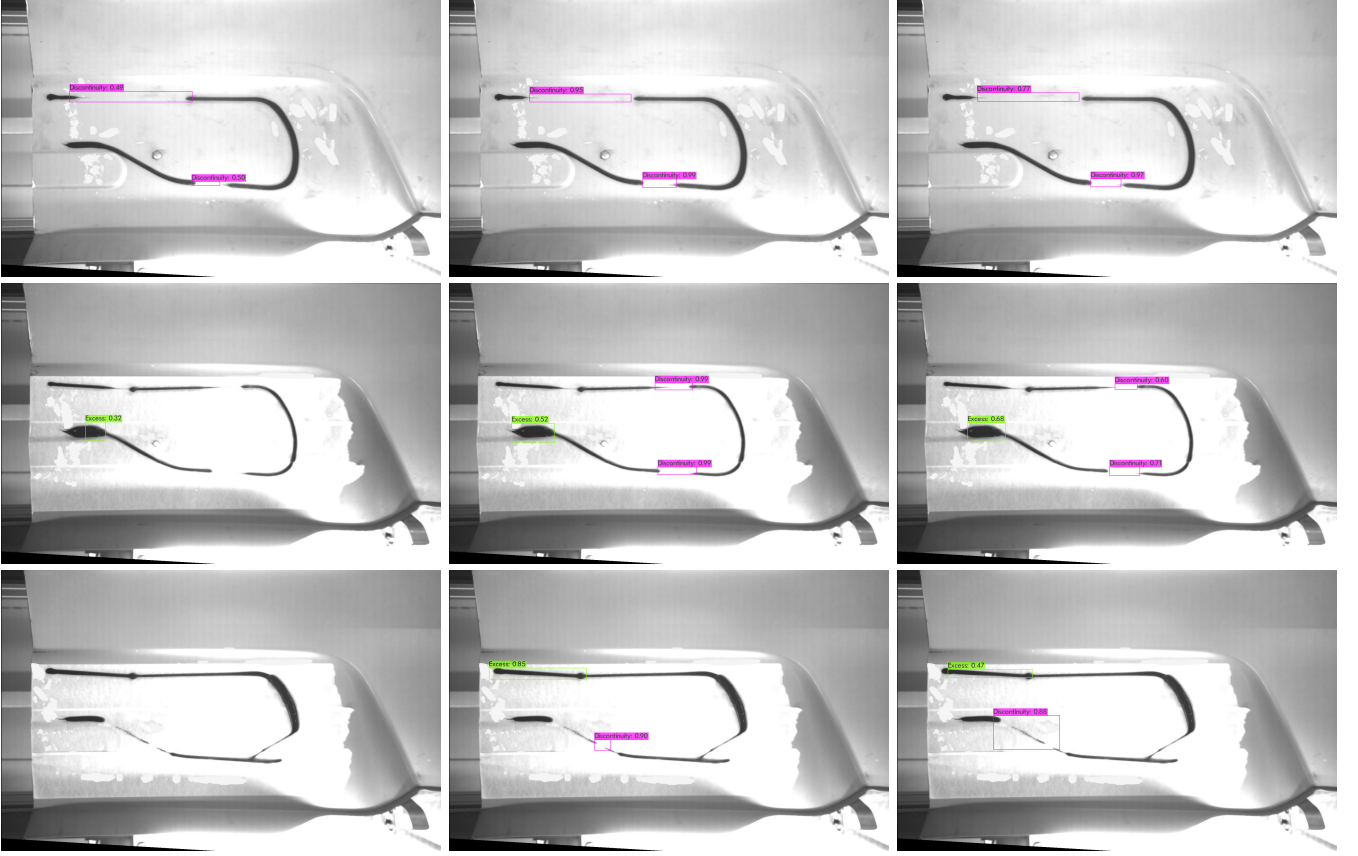


Fig. 8. Examples for the comparison between the outputs of YOLOv4 models trained on synthetic (first column), real (second column) and augmented datasets (third column) from scenario B. All variants were pre-trained on the MS COCO dataset using transfer learning.

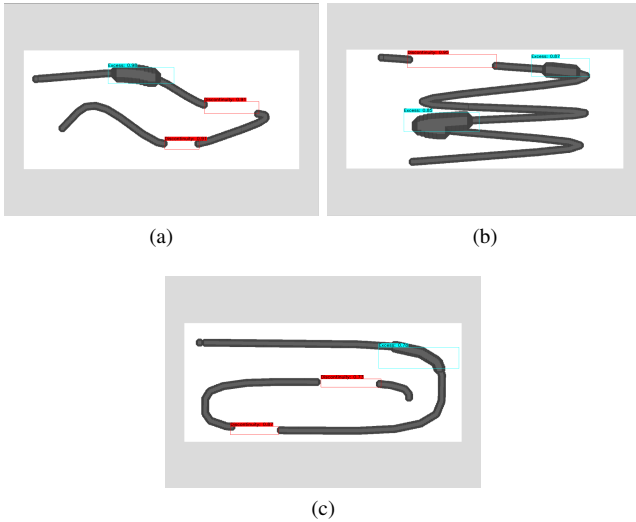


Fig. 9. Experiment to verify generalization beyond the bead shape included in the training set. Results show promise in requiring no additional effort for adaptations in the process.

approach would depend on the trade-off between modelling and annotation effort.

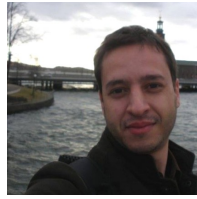
REFERENCES

- [1] W. Woyke, F. Jesche, S. Menzel, and D. Landgrebe, "Designing a flawless hem flange bonding process," *ADHESION ADHESIVES&SEALANTS*, vol. 14, no. 3, pp. 26–29, 2017.
- [2] R. S. Peres, J. Barata, P. Leitao, and G. Garcia, "Multistage quality control using machine learning in the automotive industry," *IEEE Access*, vol. 7, pp. 79 908–79 916, 2019.
- [3] C. A. Escobar and R. Morales-Menendez, "Machine learning techniques for quality control in high conformance manufacturing environment," *Advances in Mechanical Engineering*, vol. 10, no. 2, p. 1687814018755519, 2018.
- [4] S. Hinterstoisser, O. Pauly, H. Heibel, M. Martina, and M. Bokeloh, "An annotation saved is an annotation earned: Using fully synthetic training for object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.
- [5] R. S. Peres, X. Jia, J. Lee, K. Sun, A. W. Colombo, and J. Barata, "Industrial artificial intelligence in industry 4.0-systematic review, challenges and outlook," *IEEE Access*, vol. 8, pp. 220 121–220 139, 2020.
- [6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [7] D. Cabrera, F. Sancho, J. Long, R.-V. Sánchez, S. Zhang, M. Cerrada, and C. Li, "Generative adversarial networks selection approach for extremely imbalanced fault diagnosis of reciprocating machinery," *IEEE Access*, vol. 7, pp. 70 643–70 653, 2019.
- [8] W. Mao, Y. Liu, L. Ding, and Y. Li, "Imbalanced fault diagnosis of rolling bearing based on generative adversarial network: A comparative study," *IEEE Access*, vol. 7, pp. 9515–9530, 2019.
- [9] F. Zhou, S. Yang, H. Fujita, D. Chen, and C. Wen, "Deep learning fault diagnosis method based on global optimization gan for unbalanced data," *Knowledge-Based Systems*, vol. 187, p. 104837, 2020.

- [10] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, "Good semi-supervised learning that requires a bad gan," in *Advances in neural information processing systems*, 2017, pp. 6510–6520.
- [11] A. Kumar, P. Sattigeri, and T. Fletcher, "Semi-supervised learning with gans: Manifold invariance with improved inference," in *Advances in Neural Information Processing Systems*, 2017, pp. 5534–5544.
- [12] H. Di, X. Ke, Z. Peng, and Z. Dongdong, "Surface defect classification of steels with a new semi-supervised learning method," *Optics and Lasers in Engineering*, vol. 117, pp. 40–48, 2019.
- [13] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 12093–12103. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/file/8d30aa96e72440759f74bd2306c1fa3d-Paper.pdf>
- [14] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.
- [15] I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas et al., "Solving rubik's cube with a robot hand," *arXiv preprint arXiv:1910.07113*, 2019.
- [16] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2315–2324.
- [17] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," *arXiv preprint arXiv:1809.10790*, 2018.
- [18] E. Rohmer, S. P. N. Singh, and M. Freese, "Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013, www.coppeliarobotics.com.
- [19] J. Redmon, "Darknet: Open source neural networks in c," <http://pjreddie.com/darknet/>, 2013–2016.
- [20] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
- [21] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-yolov4: Scaling cross stage partial network," *arXiv preprint arXiv:2011.08036*, 2020.
- [22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [23] M. Everingham, L. Van Gool, C. K. 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, 2010.
- [24] J. Cabrita, F. Miranda, and M. Guedes, "Smart control system for zero-defect adhesive application using industrial robots," in *Workshop on Disruptive Information and Communication Technologies for Innovation and Digital Transformation*. Instituto Politécnico de Bragança Campus de Santa Apolónia 5300-253 Bragança, 2019, p. 39.



Ricardo Silva Peres received both his M.Sc. and Ph.D. in Electrical and Computer Engineering from the NOVA University of Lisbon in Portugal (2015 and 2019, respectively), specializing in the application of AI in smart manufacturing. He is currently an Invited Professor at the same university, integrating the Department of Electrical Engineering of the School of Sciences and Technology. Since 2014, he has been working as a researcher at UNINOVA - Centre of Technology and Systems focusing on the development of intelligent and predictive manufacturing systems. He has participated in several national and international research projects including FP7 PRIME, H2020 PERFORM, H2020 OpenMOS, H2020 GOOD MAN and H2020 AVANGARD. He is the author of several publications in high-ranking international scientific journals and conference proceedings (peer-reviewed). His research interests include predictive manufacturing, AI, cyber-physical systems and multi-agent systems. He has also been a member of the IEEE IES Technical Committee on Industrial Agents since 2018 and of the IEEE Standards Association P2805.1/2/3 Edge Computing Nodes Working Group since 2019.



Magno Guedes is a senior researcher, team leader and technical responsible for software development at the Innovation Department of INTROSYS. He holds a MSc. in Electrical, Electronics and Computer Engineering, from NOVA University of Lisbon, completed in 2010. He started his research career by designing and developing software applications and control systems for mobile field robots, targeting unmanned navigation in unstructured and natural environments, in the scope of national research projects such as QREN INTROBOT and QREN ROBOSAMPLER. During this work, he got specialized in machine perception, namely through computer vision. He then applied this knowledge in the industrial manufacturing sector, contributing in several projects aimed at the improvement of vision based robot guidance and human-robot collaboration (P2020 CONTIGO) and vision based quality inspection systems (P2020 SEE-Q, P2020 PROBING and P2020 CheckMate). Additionally, he has participated in international projects, including FP7 SmartAgriFood, H2020 OpenMOS, H2020 AVANGARD and H2020 DIH2.



Fábio Miranda is a senior researcher and software developer, specialized in computer vision, at the Innovation Department of INTROSYS. Fábio holds a MSc. in Electrical, Electronics and Computer Engineering from NOVA University of Lisbon, with his master's thesis entitled "Context Classification for Service Robots". He works at INTROSYS since 2015, actively participating on computer vision applications for robot guidance and quality inspection. He has participated in several national and international projects, including P2020 SEE-Q, H2020 OpenMOS, P2020 CONTIGO and P2020 CheckMate. Alongside his research career, Fábio has also been actively participating in the development of new industrial systems for non-destructive testing and quality assurance, mostly in the automotive sector.



Jose Barata is a professor at the Department of Electrical Engineering of the Nova University of Lisbon and a senior researcher of the UNINOVA Institute. He has a PhD degree in Robotics and Integrated Manufacturing since 2004 from the Nova University of Lisbon. Dr. Barata has participated in more than 15 International Research projects involving different programmes (NMP, IST, ITEA, ESPRIT). Since 2004 he has been leading the UNINOVA participation in EU projects, namely EUPASS, Self-Learning, IDEAS, PRIME, RIVER-WATCH, ROBO-PARTNER and PROSECO. His main research interests are in the area of Intelligent Manufacturing with particular focus on Complex Adaptive Systems, involving intelligent manufacturing devices. In the last years, Dr. Barata has participated actively researching SOA based approaches for the implementation of Intelligent Manufacturing Devices (e.g. within the Inlife project). He has published over 100 original papers in international journals and international conferences. He is a member of the IEEE technical committees on Industrial Agents (IES), Self-Organisation and Cybernetics for Informatics (SMC), and Education in Engineering and Industrial Technologies (IES). He is also a member of the IFAC technical committee 4.4 (Cost Oriented Automation).