Synthetic Data for Defect Segmentation on Complex Metal Surfaces

Juraj Fulir* Lovro Bosnar*† Hans Hagen† Petra Gospodnetić*
*Fraunhofer ITWM, Fraunhofer-Platz 1, 67663 Kaiserslautern
{juraj.fulir,lovro.bosnar,petra.gospodnetić}@itwm.fraunhofer.de

†RPTU Kaiserslautern-Landau, Postfach 3049, 67663 Kaiserslautern
hagen@informatik.rptu.de

Abstract

Metal defect segmentation poses a great challenge for automated inspection systems due to the complex light reflection from the surface and lack of training data. In this work we introduce a real and synthetic defect segmentation dataset pair for multi-view inspection of a metal clutch part to overcome data shortage. Model pre-training on our synthetic dataset was compared to similar inspection datasets in the literature. Two techniques are presented to increase model training efficiency and prediction coverage in darker areas of the image. Results were collected over three popular segmentation architectures to confirm superior effectiveness of synthetic data and unveil various challenges of multi-view inspection.

1. Introduction

Surface inspection is a common task of automated visual inspection systems, where defects are anomalies appearing on the surface of the product, resulting from production chain error (*e.g.* scratch, bump, pitting) [12]. While automated surface inspection systems introduce benefits such as faster inspection process and reduction of human error, they require consistent acquisition conditions. The acquisition conditions, together with detection algorithms, are always customized for specific inspection tasks. This makes the systems extremely rigid. Experts planning the acquisition hardware setup in an inspection system must ensure that the whole surface of the inspected product is illuminated and captured in a way such that all possible defects can be successfully detected. The task may sound simple in principle, however the appearance of defects varies drastically and is largely affected by the location of the defect relative to illumination sources and camera (Fig. 2). This problem becomes especially noticeable on geometrically complex and reflective metallic surfaces. Additionally, defect visibility can be obscured by the surrounding surface texture, which vary locally (Fig. 1) and between products.

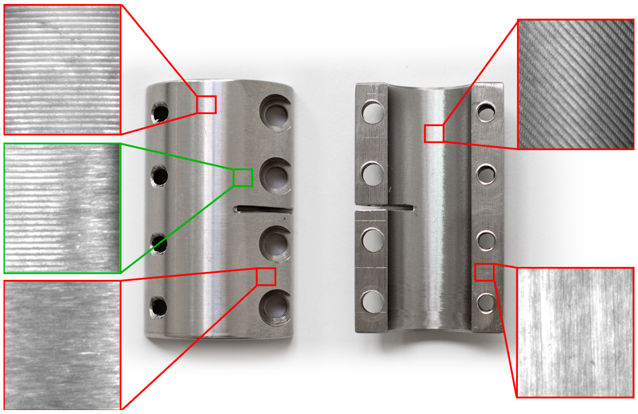

Figure 1. The examined clutch object from outside (left) and inside (right). The object contains four distinct texture patterns (red) and a transition area (green).

Designing a robust inspection system requires defect samples which are diverse enough to provide a complete understanding of all the possible defect characteristics and occurrences on the production line. A sufficient dataset will thus require a large number of physical samples, which might be challenging to obtain since some defects appear more frequently than the others and appearance within a single class of defects can vary greatly. The challenge increases for premium products fabricated in low volumes. While traditional image processing algorithms can be developed with considerably smaller amount of defected samples, cases with high variation of defect characteristics significantly complicates their development and maintenance. Machine learning approaches can circumvent these shortcomings by relying on automatic extraction of robust features from large amounts of diverse data. However, in low data scenarios they are prone to overfitting.

Usage of synthetic data to circumvent the data shortage has gained traction recently in machine vision [1, 21, 33, 39, 47, 54]. Mainly because it provides a way to generate arbitrary amount of diverse annotated training data, including edge-case scenarios which are difficult to obtain in real production. However, **there is a lack of studies which investigate the suitability and advantages of using custom designed synthetic data for industrial quality inspection**.

We summarize our contributions as following:

- We introduce a dual dataset, consisting of real and synthetic equivalent, for the domain of multi-view inspection of a complex metal object to expand the existing literature on synthetic data for industrial applications.
- We compare the effectiveness of our synthetic dataset to alternative metal inspection datasets in the literature, to confirm that a custom designed synthetic data is superior in the low-data scenario.
- We introduce intensity biased cropping mechanism to increase model training performance in this domain.
- We introduce exposure stacking to increase model response in darker regions and discuss its effect on surface coverage in inspection.
- Finally, we identify the unique shortcomings of applying synthetic data in this domain and offer research directions for overcoming them.

2. Related work

2.1. Defect recognition

Defects are the results of anomalous events in the production chain which inflict deviations from product's intended design, function, or appearance. Defects can come in various forms, with several classifications present in the literature [12, 44, 52], however, what is and what is not considered to be a defect is always application-specific. For visual inspection we distinguish between defects which are visible by observing the object with non-penetrating light interaction [12, 56], and defects that are below the observable surface and should be inspected using a material penetrating medium [2, 15, 56]. In this work we focus on the first kind, more specifically, macroscopic surface defects such as dents and scratches [12].

Recognition approaches Defect recognition is a process of identifying a defect and its characteristics. There is a number of traditional (non-learning) approaches to application specific defect recognition [10, 44]. However, these methods rely on manual design and, when presented with changes in inspection setup, require redesign which leads to an increase in complexity and operating cost. Therefore, recent research aims largely at learning based approaches where defect recognition models are trained under supervision with labeled data. The approaches can be aimed at defect detection [48, 52], defect segmentation [8, 24, 43, 52] or image classification [8, 52]. In all cases, the main problem is the high cost of obtaining large amounts of labeled data. This may increase difficulty of developing models which successfully generalize to production.

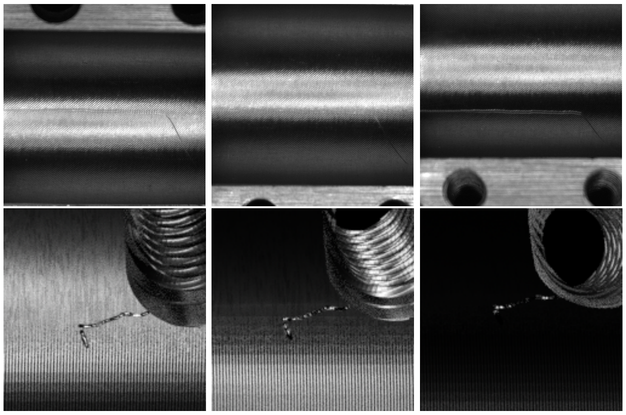

Figure 2. Appearance inconsistencies of scratches on a curved surface when acquired under different angles, as seen in real (top) and synthetic (bottom) data. Defect appearance changes in terms of its shape and contrast due to illumination and the surrounding surface texture. Notice how the defect part gets obscured by the texture in the upper middle image.

Labeling for both detection and segmentation is time consuming. Therefore, efforts have been made to use methods which rely on faster training sets annotation approaches, such as image classification [52]. There, segmentation is achieved using the class-activation map (CAM) technique [59]. CAMs tend to produce very localized predictions, which are useful for defect segmentation as defects are often localized, such as defects in LED chips [27]. Božić *et al*. [8] mixes pixel and image level labels to increase the effective dataset size at a lower annotation cost. In cases where defected samples are unavailable or are in too small quantities for supervised training, anomaly detection can be used on solely the correct samples for training. During inference, the reconstruction of an input image is compared to the original or extracted features are compared to memorized features of correct samples to detect outliers. It has been employed for defect segmentation over a variety of objects as presented in [3, 28, 36, 37]. We focus on the case where training data is scarce, making the aforementioned methods unsuitable.

Datasets Various datasets exist for defect recognition tasks in metal [42, 44, 52]. Most available datasets focus on inspection of hot-rolled steel [16, 31, 40, 41] which is a planar surface with various defects. Other present shapes are curved pads [8, 25, 43], pipes [45] or rails [58]. More complex surfaces are present in [38] in form of a ball screw driver with a multi-view setting through single-axis rotation. Anomaly detection datasets [3, 32] contain complex metal objects, however they reduce the problem to single-view inspection with a fixed top-down view. In [24], authors perform a similar top-down acquisition with varying illumination angles. In contrast, our dataset represents a complex geometry with highly specular and anisotropic surface in the multi-view setup in a dark environment.

Inspection of complex surfaces Defect recognition is tightly coupled with inspection planning process, which determines the image acquisition setup (i.e. camera and illumination position). This process is currently performed by experts based on physical tests and experience. Recently, a semi-automated inspection planning pipeline [5, 18] for virtual design and verification of inspection plans was introduced. Their work allows coverage evaluation of any object geometry, regardless of its geometrical complexity. In this work we rely on their methods to create inspection plans for both real and synthetic data.

2.2. Synthetic data generation

Image synthesis can benefit data preparation by providing more control over its content and diversity, speeding up the process and reducing its costs. Additionally, it provides insight into the expected inspection coverage and results. So far it was employed in many forms to a wide range of machine vision tasks [34] in order to produce balanced datasets for machine learning.

Generative models A straight-forward way to generate defected samples from correct real images is by synthesizing an image of a defect and embedding it in a correct real image. The cheapest approach is by manually designing generative models which produce 2D patches of defects [22]. Albeit a controllable and versatile technique, the defects are modeled as 2D patches and can not correctly model the light response of specular defects observed from multiple viewpoints. This introduces a bias towards the subset of modeled defect appearances with inaccurate light response. A more popular approach is to automatically learn the generative model using generative-adversarial networks (GAN), where two models are jointly trained in adversarial setup on weakly-annotated real data [51, 52, 57] with control over the spatial properties, category and style of defects. These approaches demonstrate great improvements in the defect recognition tasks, however they can not introduce data representing edge-case scenarios or guarantee generation of correct data. The second requirement is particularly difficult to obtain in the multi-view setup due to the complex specular defect appearance. Additionally, extending the supported set of defect types or variations requires retraining on new observed data which does not guarantee the retention of appearance quality in previously supported defect types.

Computer graphics Leveraging computer graphics for data generation provides a versatile, controllable and reliable tool for generating large quantities of data with the support for generation of scenario-specific variations. It has proven its usefulness across various computer vision tasks [13]. A popular example is traffic scene recognition where the synthetic datasets are commonly paired with real datasets [47] to complement the scenarios missing from the real data. In situations where manual data annotation is intractable, it offers an invaluable source of annotated data such as in the many-keypoint tracking task [54].

In defect recognition domain, available synthetic datasets are sparse. The DAGM dataset [53] consists of generic artificial textures and defects with no specific application in mind and is often used as a baseline benchmark. In [4], authors use simple noise transformations for color and vertex displacement to generate a labeled synthetic dataset for defect detection over steel plates. The MIAD dataset [1] is a product maintenance dataset for anomaly detection with various outdoor scenarios, including welding defects of a steel pipe. The recent CAD2Render toolkit [33] produced the DIMO dataset [14] by relying on photo-realistic rendering. It goes a bit further by using procedurally generated defects which are applied to the object surface to simulate rust and scratches. However, it focuses on assembly inspection and object pose estimation without specific control over the shape and locations of defects, reducing its usefulness for defect inspection.

In all of the above mentioned cases, the main focus is on simulating macroscopic features such as the object shape or surface color, disregarding the evaluation of the correct light response from micro-scale structures of the surface texture or the defect geometry. This is not sufficient in cases when the surface is observed from multiple viewpoints at higher resolution, reducing their usability for defect recognition. Recently, methods have been developed for generation of procedural defects [6] and procedural textures [7], capable of approximating various industrial surfaces with high degree of realism and control. The synthetic defecting methods have already been employed in [39] for defect segmentation in endoscopic images of a turbocharger.

Transfer learning Transfer learning techniques aim to align the problem domain between different data sources. A number of techniques is at hand, depending on the task and data availability [11]. These methods are suitable for use with synthetic data since the synthetic data introduces various approximations of the real world appearance, thus creating a domain shift.

Domain adaptation exploits the knowledge obtained from the source data to align the model towards the target data. The most common approach is by initializing the training procedure with model parameters pre-trained on a larger source dataset [11, 34, 47, 55]. The model can be adapted entirely [49] or partially [36]. However, some research suggests that similarity between the two domains results in better performance [11, 49].

Domain randomization [46] is a technique which enlarges the variance of the source domain to increase the chance of covering the target domain, while making it possible for the model to learn more robust features. It is commonly used in synthetic data since the environment can be easily manipulated [34, 47]. By parameterizing the defect geometry and surface texture as procedural functions [6, 7], we can generate a variety of data that can cover all plausible possibilities within the specified ranges. Note that some parameters, such as perspective distortion, rotation or flipping of the image, are commonly randomized through train-time augmentation removing the need for their rendering.

Following Wood *et al*. [54] instead of using domain adaptation to reduce the domain gap, we rely on increasing the realism of synthetic data. However, we do incorporate domain randomization of surface appearance and background to produce a variety of realistic samples.

3. The clutch dataset

In this work we introduce and publish[1] the dual dataset composed from real data and its synthetic equivalent. It is a versatile dataset which can be utilized for multiple tasks such as image classification, defect segmentation and detection in supervised, unsupervised or weakly-supervised approaches. In this work we focus on binary defect segmentation in order to evaluate approaches for dataset preparation, machine learning techniques specialized for this domain and possible imperfections of synthetic data which must be taken into account.

3.1. Object description

The dataset contains a part of a clutch, shown in Fig. 1. The clutch is an aluminum object consisting of two halves, produced using turning and milling with additional brushing to remove material extrusions introduced from drilling. The flat and curved surfaces, holes and details such as screw threads or beveled edges increase the geometrical complexity of the object. Different machining and processing operations throughout the part production introduce four distinct surface textures displaying patterns with more or less prominent periodicity.

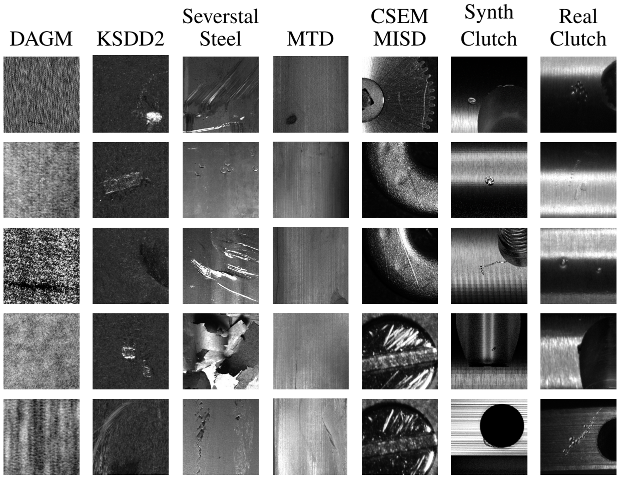

Figure 3. Texture and defect examples extracted from datasets used in this paper. Best viewed digitally.

Since the real defects are often a proprietary information, the clutch object is used as a case study and the defects were introduced manually to resemble typical defects appearing in production lines. The defects include various scratches and dents depicted in Fig. 3.

3.2. Real data acquisition

The RealClutch acquisition setup consists of a robot manipulator, matrix grayscale camera with a diffuse ring light mounted around it and the acquisition table. The manipulator is used to position the camera and the illumination into predefined viewpoints. The acquisition table is a flat surface covered in diffuse black velvet and the inspected object is placed on it. The viewpoints were arranged manually using V-POI[2] [20] in a way that covers the inspected surfaces with overlaps [19]. For the purpose of this work, the viewpoints have been created with significant overlap in order to examine defect behavior from multiple acquisition angles. Before the acquisition, hand-to-eye calibration has been performed as in [5], however slight acquisition offsets are still present due to manual object placement on the acquisition table. The collected images were manually annotated using *labelme* [50] with an extension which allows enhancement of the defect visibility (Fig. 4) by manually adjusting the image exposure using: $f(x) = x \cdot 2^{\alpha}$, with $\alpha \in \{0, 1, 2\}$. The polygonal annotations were finally rasterized into image masks used for model training and validation.

3.3. Synthetic data generation

The SynthClutch dataset has been designed and generated using the methods presented by Bosnar *et al*. in [5–7]. The process requires a 3D model of the object and consists

[1] https://owncloud.fraunhofer.de/index.php/s/mtr1FzERutdOrXi

[2] https://itwm.fraunhofer.de/v-poi

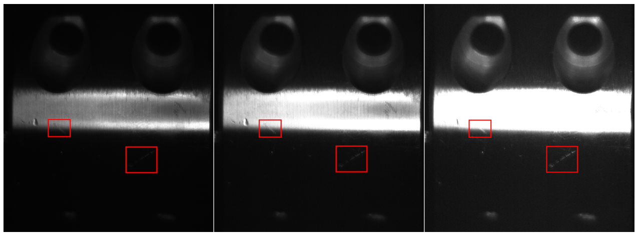

Figure 4. Increasing image exposure reveals the dark, but also overexposes the bright parts of the object. Notice the appearance change of the lower right scratch and the middle left scratch.

of four steps: inspection planning, defect modeling, texture definition and dataset generation. In the inspection planning step, view and illumination points are positioned in the space relative to the object. The viewpoints contain camera parameters such as resolution, focal length or focusing distance and illumination points contain the light geometry and intensity. For the purpose of this work, all the parameters correspond to the real setup, the lightpoint geometry has the ring light shape and is positioned around the viewpoint. The defects are modeled as dents and scratches, imprinted directly into the object geometry. They are defined using class specific parameter ranges, sampled to obtain desired shape variation. Dents are defined by their size, depth and elongation, while scratches are defined by their depth, length, and curving strength and frequency. The textures are modeled using procedural methods that perturb surface normals, with parameters adjusted to match the observed appearance of their respective counterparts across the real object (Fig. 1). Finally, all aforementioned elements are combined and parameters are sampled to define a scene for photo-realistic rendering of the images and their corresponding defect masks, rendered using emissive material. As the real environment contains inter-reflections between the object and acquisition manipulator, we emulate this with a small amount of constant illumination. The background was dark with addition of uniform noise at train-time to resemble sensor dark shot noise.

The dataset contains object instances varying in defect shapes and texture which are then rendered according to the inspection plan. First, geometry instances are created with defects randomly generated and applied across the surface. The defect sizes were defined to be comparable to defects present in real samples and contain circular dents ($\varnothing 0.2{-}2mm$) and scratches ($\varnothing 0.05 - 0.3mm$). Additionally, we apply insignificant defects which do not contribute to the masks, but simulate minor irregularities present in the real data which may resemble defects but should be ignored. Next, the texture parameters are sampled within defined ranges centered around their previously optimized values. We randomize the surface roughness, normal perturbation strength and texture scale. Finally, each geometry instance is rendered with its corresponding texture parameters and stored in a structured way to simplify data loading. Reader is referred to the supplementary for a visual comparison of the two datasets.

4. Defect segmentation on complex surfaces

Complex metal objects present a unique case for defect recognition due to the changes in surface appearance caused by reflectivity. This requires collection of large amounts of data to successfully train a model, however oftentimes this is not possible. In our case, only a small number of real samples is available real making it difficult to restrain models from overfitting, even with extensive data augmentation. Therefore, we look into using alternative data sources from similar domains and compare its use to a custom designed synthetic dataset. In both cases the real data is utilized only for fine-tuning and evaluation.

4.1. Utilizing existing planar datasets

We first examine the transfer of features learned on available large datasets that represent similar domains. These datasets collect a large number of images with a variety of defect types observed over mostly planar surfaces. We restrict our selection to 5 datasets (Fig. 3) based on similarities to RealClutch: DAGM [53] for its genericness, KSDD2 [8] for its defect shapes, Severstal Steel [40] and MTD [25] for their material and defects, and CSEM-MISD [24] for its materials, defects and defect visibility changes.

4.2. Utilizing custom designed synthetic data

Synthetic data can be generated in arbitrary amounts with large diversity. However, it introduces a domain gap due to the approximations made during simulation such as texture geometry or material reflectance. We analyze the applicability of features learned on synthetic data with and without fine-tuning on the real data.

Intensity-biased random cropping Inspection images have larger resolution to increase surface coverage, which requires us to use random cropping during training. As large part of the image is in the dark, random cropping produces a large amount of crops that end up in the dark background regions. To remedy this, we bias the random cropping mechanism towards brighter regions to contain the object's surface. The input image is first binarized using a user-defined threshold to obtain an intensity mask. A random pixel from the mask is sampled and a crop window is centered around it. This technique guarantees that every crop will contain useful intensity values, while not completely ignoring the darker regions to prevent the possibility of detecting false positives in the acquisition scene background. The threshold was chosen heuristically from real data to ensure coverage of the regions containing manual annotations.

4.3. Enhancing model response in dark regions

When predicting on the acquired image, the model tends to respond only on well lit surfaces where the patterns have higher contrast, which reduces the detection coverage over the object surface. As described in Sec. 3.2 annotators could increase image exposure to enhance the visibility of defect shapes in darker areas of the image, at the cost of overexposing some areas and increasing the image noise amplitude (Fig. 4). To emulate this, we transform the acquired image using same exposure values as annotators to construct a channel-wise stack of transformed images alongside the original as inputs to the model. This allows our model to have simultaneous access to multiple exposure values and learn to respond to a much larger surface area.

5. Experimental evaluation

5.1. Training details

Training is implemented using PyTorch and for segmentation techniques we used easily accessible implementations of segmentation models: FCN [29] and DeepLabV3 [9] implementations from torchvision and U-Net [35] from segmentation-models library [26], with the ResNet-34 [23] backbone. For FCN and DeepLabV3 backbones we additionally use bottleneck blocks to increase speed [23] and dilated convolutions in the last 3 layers to keep the resolution reduction factor to 8 [9]. The dilation was necessary to obtain precise predictions.

For training we use AdamW [30] and binary cross-entropy (BCE) loss function. To satisfy memory constraints we use random crops of size 256×256, biased towards intensity values above 10. We find that further lowering of crop sizes decreases model performance. The use of intensity biased cropping in most of the cases reduced the training time and produced baseline models with metrics increased by few percentage points. Therefore, it was employed in all of the experiments. We train with batch size 16 for a maximum of 1000 epochs, selected from preliminary experiments on RealClutch. The initial learning rate and L2 weight decay factor were always selected using grid search over $\{10^{-3}, 10^{-4}\}$ and $\{10^{-4}, 10^{-5}\}$ respectively, maximizing F1 score on source validation set. The learning rate was halved every 50 epochs, with early stopping when relative decrease in validation loss is under 0.01 for 5 consecutive validations. Model parameters pre-trained on ImageNet did not improve speed or performance, similarly to [49]. Once the model is trained, fine-tuning is performed using real clutch images. For fine-tuning we only reduce the starting learning rate to 10^{-4} and train until convergence of validation loss. In all experiments, image values were centered to range $[-1, 1]$ and the defect mask channels were collapsed for single class segmentation. The reader is referred to the supplementary for detailed analysis.

The performance of our models and data sources is compared using pixel-wise metrics: precision, recall and F1 score. The model predictions were binarized with a threshold that maximizes the F1 score on validation set of the respective training dataset.

The RealClutch dataset consists of 3 correct and 3 defected objects. The objects were acquired using 86 viewpoints, covering all examined surfaces, resulting in 516 labeled images of resolution 2448×1025. The train-test split was constructed object-wise using 2 and 4 objects respectively, while keeping the 1:1 balance between the correct and defected samples. The train set contains objects with one surface texture, while the test set contains two surface textures which is common in evolving manufacturing processes. The train-val split was constructed with a 4:1 random split of the training set. The image resolution is halved for evaluation efficiency, padded to ensure divisibility by 96 and split into patches of size 416×352.

5.2. Effectiveness of planar datasets

When evaluating the usefulness of pre-training on existing datasets, the domain difference is compensated for using augmentations. We apply random rotations from $[-90, 90]$, Gaussian noise of variance ≤ 10, exposures between $[0, 1]$, Gaussian blur with kernel sizes form $\{1, 3\}$, horizontal and vertical flips. For each dataset the model was trained on the source dataset and evaluated on the RealClutch test set with and without fine-tuning.

DAGM [53] is a synthetic dataset of generic textures with artifacts representing defects. It consists of 10 textures, totaling with around 8000 labeled samples for train and test splits. The labels are in form of ellipsoids surrounding the defected area which does not provide a detailed coverage of the defect and is a form of weak supervision. We pad the images to size 512 and split into patches of size 256×256.

Kolektor Surface Defect Detection v2 (KSDD2) [8] is a dataset for binary defect segmentation of metallic tile-shaped products with rough surface. The train split contains 2331 samples, which we split in 4:1 ratio for training and validation. The test split contains 1004 samples. For evaluation efficiency, we convert the images to grayscale, pad them with zeros to resolution 256×672 and split into patches of size 256×224.

Severstal Steel dataset [40] is a dataset of planar hot-rolled steel with defects segmented in 4 classes which we collapse into binary segmentation. The test set for this dataset is private, however we use it solely for the pre-training of models so we utilize the available train split for training and validation. The train split contains 12568 samples, which we split using 4:1 ratio for training and validation respectively. We merge the defect classes into single class to make it compatible with our binary segmentation. For evaluation efficiency, we pad the images with ze-

Source dataset	FCN			DLv3			U-Net		
	P [%]	R [%]	F1 [%]	P [%]	R [%]	F1 [%]	P [%]	R [%]	F1 [%]
RealClutch (baseline)	53.2	19.4	28.4	59.1	16.5	25.8	55.7	21.7	31.3
DAGM	0.0	4.6	0.1	1.0	0.1	0.1	0.1	5.9	0.2
KSDD2	1.5	5.1	2.3	2.2	6.5	3.3	0.7	7.0	1.3
Severstal Steel	1.0	9.7	1.8	1.7	9.0	2.9	1.0	3.2	1.5
MTD	7.7	10.2	8.8	23.5	9.9	13.9	8.7	11.6	10.0
CSEM-MISD	6.5	6.9	6.7	9.3	3.8	5.4	4.6	5.0	4.8
DAGM (FT)	15.5	4.7	7.2	9.3	5.3	6.7	20.0	3.3	5.6
KSDD2 (FT)	8.0	3.9	5.3	2.2	1.2	1.5	3.6	0.8	1.3
Severstal Steel (FT)	33.7	12.9	18.6	19.5	8.6	11.9	6.1	12.0	8.0
MTD (FT)	28.8	10.5	15.3	48.1	6.6	11.5	48.0	9.5	15.9
CSEM-MISD (FT)	55.2	17.5	26.5	55.0	19.9	29.2	46.3	13.5	20.9
SynthClutch	59.3	10.7	18.1	57.4	10.7	18.1	67.2	10.8	18.7
SynthClutch (FT)	**69.6**	24.0	**35.7**	63.1	25.5	36.3	**67.6**	**28.9**	**40.5**
RealClutch (EX)	59.0	16.3	25.5	54.8	17.9	27.0	55.2	20.3	29.7
SynthClutch (EX)	58.1	11.6	19.4	57.5	12.0	19.8	60.0	11.6	19.4
SynthClutch (EX+FT)	67.9	**24.2**	**35.7**	**67.6**	**27.7**	**39.3**	64.9	23.5	34.6

Table 1. Comparison between models trained on different source datasets, including fine-tuning (FT) and exposure stacking (EX), evaluated on RealClutch test split. Precision (P), recall (R) and F1 score (F1) are presented. The best results are bolded vertically.

ros to resolution 1792×256 and split it into patches of size 224×256.

Magnetic Tile Defects (MTD) [25] is a dataset of magnetic tiles with slight curvature, used for saliency prediction over 5 defect classes. We select the defect classes that are important for our task based on their similarity to our data. We treat *blowhole*, *crack* and *break* as the defected class, while *uneven* and *free* are ignored. The *fray* class is not expected in our data and is thus ignored. Our subset of this dataset contains 1312 samples split into train-val sets using the 4:1 ratio, while keeping the ratio of correct and defected samples at 4:1 in both subsets. For evaluation efficiency, we standardize the resolution to 640×448 by padding with zeros and split the image into patches of size 320×224.

CSEM multi-illumination surface defects (CSEM-MISD) [24] collects 3 different objects (gear, screw and washer) used for defect segmentation. We use the highest 24 light points as defined in the paper, totaling to 2304 images. Different from the proposed method, we train the model to predict defects on each image separately. The train split contains 32 instances of every object which we split using 4:1 ratio for training and validation. For evaluation efficiency, we split the image into patches of size 256×256.

In Tab. 1 generalization capabilities of models trained on different source datasets to the baseline model trained on RealClutch. As expected, the models trained on the RealClutch data generalize poorly due to the small number of available samples and overfitting due to training and validation being performed on different images but same objects. DAGM performs even worse since it does not model neither the surface texture nor the defect appearance of the metallic surfaces nor the tight segmentation masks. Severstal Steel shows some promise due to its size and defect variety which helps in regularizing the model to learn more robust features, which is especially visible after fine-tuning. MTD is most similar to surfaces of RealClutch and the results confirm this relative to other sources. CSEM-MISD additionally displays changes in defect appearance and after fine-tuning performs on par with RealClutch. Fine-tuning the models with RealClutch in most cases causes a significant increase of performance, with highest performance change attributed to the most similar datasets. However these gains cannot be predicted based on the pre-trained model performance. This allows the conclusion that the domain similarity in form of surface and defect appearance is important for knowledge transfer. Additionally, learning to correctly respond to different surface and defect appearances is a crucial information for better performance.

5.3. Effectiveness of custom designed synthetic data

SynthClutch dataset consists of 20 correct and 20 defected object instances. We used the same viewpoints as for the real acquisition including the non-examined surfaces, totaling in 106 viewpoints and 4240 labeled images. The train-val-test split was constructed object-wise using $28-4-8$ objects respectively, while keeping the balance between correct and defected objects. We follow the augmentation from Sec. 5.2, with max rotation angle reduced to 30 to avoid learning out-of-domain features.

Compared to the baseline model, the models trained on synthetic data produce lower recall and similar precision. However, fine-tuning on RealClutch boosts the performance above the baseline models by $5-10\%$ on both metrics. Most pronounced increase is in recall, where the model mostly increased the area of predictions to match the labels with a few additional defects becoming detected. When compared to pre-training on planar datasets, synthetic data doubles the model performance. This shows that task specific features guided by geometric attributes and surface texture are required for best prediction quality. Consequently, this simplifies the task for fine-tuning as model needs to adapt only to the smaller differences between domains.

The exposure stacking augmentation is evaluated only on SynthClutch since the defect behavior corresponds to the target RealClutch defects. As expected, in most cases recall increases as the model becomes more responsive to a larger surface area. However, the results are not consistently better or worse, indicating a need for more detailed research.

6. Discussion

Existing planar datasets were of limited value as sources of data for transferring knowledge to our geometrically complex domain. The real object contains sharp and curved geometrical features with tiny insignificant defects which can appear very bright under different views. Models tend to produce false-positives in those regions as they were not explicitly trained to ignore them. Even fine-tuning this does not fully resolve this issue, raising the importance of training on the target object data from the start.

The use of custom designed **synthetic data has proven to be the most promising approach when the amount of real data is extremely restricted**. Although still hindered by the domain gap, the overall model performance is significantly better than using models trained on alternative datasets, which in many cases contain more training samples but the domain is not similar enough. The model performance is additionally impeded by the task difficulty. Significant appearance changes of defects depending on the grazing angle and surrounding texture produce ambiguity which would also be present for the human inspector. In such cases, a human inspector would not make a conclusion, but seek a different grazing angle, which was mimicked by the illumination stacking approach of Honzatko *et al.* [24]. In automated inspection the network is expected to decide based on a single view, which lowers the recall rate as observed in Tab. 1. This comes from the fact that the synthetic data is overly precisely labeled - the defects are labeled if they are geometrically visible (not obstructed), and not if they are visible in terms of prominence. This is true for [24] as well. This issue hints at the need for models with efficient multi-view memory capabilities or models utilizing the grazing angle and location information about the view from a CAD model. So far, the research community has no answer to defect visibility evaluation, however our study raises this as an important point to be tackled in the future to prevent over-labeling in synthetic data.

The defect visibility problem is also closely related to the estimation of inspection coverage where we estimate if a region of the object surface can be inspected by a set of viewpoints. Our study on multiple exposures unveils the opportunity to study the *effective visual coverage* that is achieved by a particular recognition model, which greatly influences the process of inspection planning [19].

While fine-tuning on a small amount of real data helps with the domain gap, it is still prone to model overfitting due to high complexity of both the task and the model. [17] Further development of the data generator could reduce the gap from reality and reduce the need for fine-tuning. Enhancements might include texture models with richer variations or more precise selection of texture parameters. Both is achievable due to the immense controllability of generators based on computer graphics. Once designed, the data generator can reuse existing textures and defects, and be extended for the new ones. The extensions may be costly in terms of time however they become cheaper on the long run due to their high reusability and adaptability across different inspection targets.

7. Conclusion

When it comes to metal inspection, weighting the benefits of investing into a custom designed synthetic dataset against using publicly available datasets is difficult. Metal as a target domain is alone highly restrictive, whereas the multi-view inspection of complex metal geometry leaves us with a single publicly available real dataset. Therefore this work not only examined the benefits of the synthetic data, but additionally published a new dataset containing both real and corresponding synthetic data for multi-view inspection of a complex metal object. Such dataset is a first of its kind for metal inspection. The synthetic data has proven to be a superior pre-training data source over multiple architectures but is still burdened by over-labeling. To resolve this, the research must further focus on generator enhancement, defect visibility quantification and utilization of object 3D as additional source of information for the network.

8. Acknowledgment

This work was supported by Fraunhofer ITWM and the German Federal Ministry of Education and Research (BMBF) [grant number 01IS21058B (SynosIs)]. We thank Maedler for providing us with the correct object samples and consent to use them for research.

References

[1] Tianpeng Bao, Jiadong Chen, Wei Li, Xiang Wang, Jingjing Fei, Liwei Wu, Rui Zhao, and Ye Zheng. Miad: A maintenance inspection dataset for unsupervised anomaly detection, 2022. 1, 3

[2] Tin Barisin, Christian Jung, Franziska Müsebeck, Claudia Redenbach, and Katja Schladitz. Methods for segmenting cracks in 3d images of concrete: A comparison based on semi-synthetic images. *Pattern Recognition*, 129:108747, 2022. 2

[3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad — a comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9584–9592, 2019. 2, 3

[4] Aleksei Boikov, Vladimir Payor, Roman Savelev, and Alexandr Kolesnikov. Synthetic data generation for steel defect detection and classification using deep learning. *Symmetry*, 13(7), 2021. 3

[5] Lovro Bosnar, Siddhartha Dutta, Doria Saric, Thomas Weibel, Markus Rauhut, Hans Hagen, and Petra Gospodnetić. Image synthesis pipeline for surface inspection. In *LEVIA'20: Leipzig Symposium on Visualization in Applications*, 2020. 3, 4

[6] Lovro Bosnar, Hans Hagen, and Petra Gospodnetić. Procedural defect modeling for virtual surface inspection environments. *IEEE Computer Graphics and Applications*, 2023. 3, 4

[7] Lovro Bosnar, Markus Rauhut, Hans Hagen, and Petra Gospodnetic. Texture synthesis for surface inspection. In *LEVIA 22: Leipzig Symposium on Visualization in Applications.*, 2022. 3, 4

[8] Jakob Božič, Domen Tabernik, and Danijel Skočaj. Mixed supervision for surface-defect detection: from weakly to fully supervised learning. *Computers in Industry*, 2021. 2, 3, 5, 6

[9] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv e-prints*, page arXiv:1706.05587, June 2017. 6

[10] Yajun Chen, Yuanyuan Ding, Fan Zhao, Erhu Zhang, Zhangnan Wu, and Linhao Shao. Surface defect detection methods for industrial products: A review. *Applied Sciences*, 11(16), 2021. 2

[11] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey, 2017. 3, 4

[12] Tamás Czimmermann, Gastone Ciuti, Mario Milazzo, Marcello Chiurazzi, Stefano Roccella, Calogero Maria Oddo, and Paolo Dario. Visual-based defect detection and classification approaches for industrial applications—a survey. *Sensors*, 20(5), 2020. 1, 2

[13] Tim Dahmen, Patrick Trampert, Faysal Boughorbel, Janis Sprenger, Matthias Klusch, Klaus Fischer, Christian Kübel, and Philipp Slusallek. Digital reality: a model-based approach to supervised learning from synthetic data. *AI Perspectives*, 1(1):Article No.2, 2019. 43.22.02; LK 01. 3

[14] Peter De Roovere, Steven Moonen, Nick Michiels, and Francis Wyffels. Dataset of industrial metal objects, 2022. 3

[15] Qiang Fang, Clemente Ibarra-Castanedo, and Xavier Maldague. Automatic defects segmentation and identification by deep learning algorithm with pulsed thermography: Synthetic and experimental data. *Big Data and Cognitive Computing*, 5(1), 2021. 2

[16] Guizhong Fu, Peize Sun, Wenbin Zhu, Jiangxin Yang, Yanlong Cao, Michael Ying Yang, and Yanpeng Cao. A deep-learning-based approach for fast and robust steel surface defects classification. *Optics and Lasers in Engineering*, 121:397–405, 10 2019. 2

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org. 8

[18] Petra Gospodnetić. *Visual Surface Inspection Planning for Industrial Applications*. PhD thesis, 2022. 3

[19] Petra Gospodnetić, Dennis Mosbach, Markus Rauhut, and Hans Hagen. Viewpoint placement for inspection planning. 33(2), 2022. 4, 8

[20] Petra Gospodnetić, Markus Rauhut, and Hans Hagen. Surface inspection planning using 3d visualization. In *LEVIA'19: Leipzig Symposium on Visualization in Applications*, 2019. 4

[21] Xi Guo, Wei Wu, Dongliang Wang, Jing Su, Haisheng Su, Weihao Gan, Jian Huang, and Qin Yang. Learning video representations of human motion from synthetic data. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20165–20175, 2022. 1

[22] Matthias Haselmann and Dieter Gruber. Supervised machine learning based surface inspection by synthetizing artificial defects. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 390–395, 2017. 3

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 6

[24] David Honzátko, Engin Türetken, Siavash A. Bigdeli, L. Andrea Dunbar, and Pascal Fua. Defect segmentation for multi-illumination quality control systems. *Machine Vision and Applications*, 32(6):118, Sep 2021. 2, 3, 5, 7, 8

[25] Yibin Huang, Congying Qiu, Yue Guo, Xiaonan Wang, and Kui Yuan. Surface defect saliency of magnetic tile. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, pages 612–617, 2018. 3, 5, 7

[26] Pavel Iakubovskii. Segmentation models. https://github.com/qubvel/segmentation_models, 2019. 6

[27] Hui Lin, Bin Li, Xinggang Wang, Yufeng Shu, and Shuanglong Niu. Automated defect inspection of led chip using deep convolutional neural network. *Journal of Intelligent Manufacturing*, 30:1–10, 08 2019. 2

[28] Jiaqi Liu, Guoyang Xie, Jingbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. Deep industrial image anomaly detection: A survey, 2023. 2

[29] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, Los Alamitos, CA, USA, jun 2015. IEEE Computer Society. 6

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. 6

[31] Xiaoming Lv, Fajie Duan, Jia-jia Jiang, Xiao Fu, and Lin Gan. Deep metallic surface defect detection: The new benchmark and detection network. *Sensors*, 20(6), 2020. 2

[32] Pankaj Mishra, Riccardo Verk, Daniele Fornasier, Claudio Piciarelli, and Gian Luca Foresti. VT-ADL: A vision transformer network for image anomaly detection and localization. In *30th IEEE/IES International Symposium on Industrial Electronics (ISIE)*, June 2021. 3

[33] Steven Moonen, Bram Vanherle, Joris de Hoog, Taoufik Bourgana, Abdellatif Bey-Temsamani, and Nick Michiels. CAD2Render: A modular toolkit for GPU-accelerated photorealistic synthetic data generation for the manufacturing industry. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 583–592, January 2023. 1, 3

[34] Sergey I. Nikolenko. Synthetic data for deep learning, 2019. 3, 4

[35] O. Ronneberger, P.Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 6

[36] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14298–14308, 2022. 2, 4

[37] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2592–2602, January 2023. 2

[38] Tobias Schlagenhauf, Magnus Landwehr, and Jürgen Fleischer. Industrial machine tool element surface defect dataset, 2021. 3

[39] Ole Schmedemann, Melvin Baaß, Daniel Schoepflin, and Thorsten Schüppstuhl. Procedural synthetic training data generation for ai-based defect detection in industrial surface inspection. *Procedia CIRP*, 107:1101–1106, 2022. Leading manufacturing systems transformation – Proceedings of the 55th CIRP Conference on Manufacturing Systems 2022. 1, 3

[40] PAO Severstal. Severstal: Steel defect detection. https://www.kaggle.com/c/severstal-steel-defect-detection, 2019. 2, 5, 6

[41] Kechen Song and Yunhui Yan. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. *Applied Surface Science*, 285:858–864, 11 2013. 2

[42] Xiaohong Sun, Jinan Gu, Shixi Tang, and Jing Li. Research progress of visual inspection technology of steel products—a review. *Applied Sciences*, 8(11), 2018. 2

[43] Domen Tabernik, Samo Šela, Jure Skvarč, and Danijel Skočaj. Segmentation-Based Deep-Learning Approach for Surface-Defect Detection. *Journal of Intelligent Manufacturing*, May 2019. 2, 3

[44] Bo Tang, Li Chen, Wei Sun, and Zhong-kang Lin. Review of surface defect detection of steel products based on machine vision. *IET Image Processing*, 17(2):303–322, 2023. 2

[45] Tianchi. Aluminum profile surface flaw recognition dataset, 2016. 3

[46] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter 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)*, pages 23–30, 2017. 4

[47] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1082–10828, Los Alamitos, CA, USA, jun 2018. IEEE Computer Society. 1, 3, 4

[48] Andrei-Alexandru Tulbure, Adrian-Alexandru Tulbure, and Eva-Henrietta Dulf. A review on modern defect detection models using dcnns – deep convolutional neural networks. *Journal of Advanced Research*, 35:33–48, 2022. 2

[49] Bram Vanherle, Steven Moonen, Frank Van Reeth, and Nick Michiels. Analysis of training object detection models with synthetic data. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022. 4, 6

[50] Kentaro Wada. Labelme: Image polygonal annotation with Python. 4

[51] Ruyu Wang, Sabrina Hoppe, Eduardo Monari, and Marco Huber. Defect transfer gan: Diverse defect synthesis for data augmentation. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022. 3

[52] Xin Wen, Jvran Shan, Yu He, and Kechen Song. Steel surface defect recognition: A survey. *Coatings*, 13(1), 2023. 2, 3

[53] Matthias Wieler, Tobias Hahn, and Fred. A. Hamprecht. Weakly supervised learning for industrial optical inspection [dataset]. https://hci.iwr.uni-heidelberg.de/content/weakly-supervised-learning-industrial-optical-inspection, 2007. 3, 5, 6

[54] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J. Cashman, and Jamie Shotton. Fake it till you make it: Face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3681–3691, October 2021. 1, 3, 4

[55] Chengzhi Wu, Xuelei Bi, Julius Pfrommer, Alexander Cebulla, Simon Mangold, and Jürgen Beyerer. Sim2real trans-

fer learning for point cloud segmentation: An industrial application case on autonomous disassembly. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4531–4540, January 2023. 4

[56] Jing Yang, Shaobo Li, Zheng Wang, Hao Dong, Jun Wang, and Shihao Tang. Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. *Materials*, 13(24), 2020. 2

[57] Gongjie Zhang, Kaiwen Cui, Tzu-Yi Hung, and Shijian Lu. Defect-GAN: High-fidelity defect synthesis for automated defect inspection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2524–2534, January 2021. 3

[58] Zihao Zhang, Shaozuo Yu, Siwei Yang, Yu Zhou, and Bingchen Zhao. Rail-5k: a real-world dataset for rail surface defects detection [unpublished]. *CoRR*, abs/2106.14366, 2021. 3

[59] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2