Generative Adversarial Network Applications in Industry 4.0: A Review

Chafic Abou Akar[1,2*], Rachelle Abdel Massih[1], Anthony Yaghi[1,2], Joe Khalil[1,2], Marc Kamradt[1] and Abdallah Makhoul[2]

[1*]BMW Group, Munich, Germany.
[2]Université de Franche-Comté, CNRS, institut FEMTO-ST, 25200 Montbéliard, France.

*Corresponding author(s). E-mail(s): chafic.ac.abou-akar@bmw.de;
Contributing authors: rachelle.abdel-massih@bmw.de; anthony.yaghi@bmw.de; joe.khalil@bmw.de; marc.kamradt@bmw.de; abdallah.makhoul@univ-fcomte.fr;

Abstract

The breakthrough brought by generative adversarial networks (GANs) in computer vision (CV) applications has gained a lot of attention in different fields due to their ability to capture the distribution of a dataset and generate high-quality similar images. From one side, this technology has been rapidly adopted as an alternative to traditional applications and introduced novel perspectives in data augmentation, domain transfer, image expansion, image restoration, image segmentation, and super-resolution. From another side, we found that due to the lack of industrial datasets and the limitation for acquiring and accurately annotating new images, GANs form an exciting solution to generate new industrial image datasets or to restore and augment existing ones. Therefore, we introduce a review of the latest trend in GANs applications and project them in industrial use cases. We conducted our experiments with synthetic images and analyzed most of GAN's failures and image artifacts to provide training's best practices.

Keywords: Computer Vision, Generative Adversarial Networks, Image Generation, Image Translation, Industry 4.0

Acknowledgments. This research is supported by the EIPHI Graduate School (contract "ANR-17-EURE-0002") and the BMW TechOffice Munich.

Statements and Declarations

Conflict of interest. The authors declare that they have no conflict of interest.

1 Introduction

Obtaining sufficient training images is a major obstacle in deep learning (DL) and industrial computer vision (CV). While transfer learning (TL) has helped to alleviate this issue through publicly available high-quality datasets, the size of available industrial datasets is still smaller compared to publicly available CV datasets. This is due to security, privacy, and robustness constraints for the acquirer, employees, and the project. Additionally, traditional techniques for augmenting datasets,

such as geometric, color, or kernel filter transformations, may not be suitable for specific industrial CV models. The industrial sector has specific standards and conventions, such as box dimensions, color codes, and sign orientations, that must be considered while augmenting. Acquiring data becomes challenging when considering the manual labor required to annotate such a huge dataset for supervised training, such as bounding boxes, segmentation, depth information, and pairing images. This task is prone to human error, time-consuming, subjective, and costly because of the complexity and criticality of the images and the need for specialists in the industrial field.

Generative Adversarial Network (GAN) is a training methodology that can augment existing image datasets by producing high-quality synthetic images. The concept of GANs was first proposed by Goodfellow *et al.* (Goodfellow et al, 2014). GANs consist of two adversarial deep neural networks (DNNs): a generator G and a discriminator D. These networks are trained in an adversarial zero-sum game to find the Nash equilibrium, where G captures the distribution of the dataset and produces data that appears to be real to confuse D. In turn, D is trained to estimate the probability of accurately distinguishing, as genuinely as possible, whether the data is from a real distribution or generated by G.

Hence, GANs are a powerful approach to generating realistic images using backpropagation techniques. This method is more effective than traditional deep generative models such as Boltzmann machines, variational autoencoders (VAEs), and Markov chain-based algorithms (Chen and Jia, 2021). This has led to GANs being widely adopted and applied in various real-world applications, including the medical field (Yi et al, 2019), molecular imaging (Koshino et al, 2021), deep fake, cybersecurity (Yinka-Banjo and Ugot, 2020; Cai et al, 2021), finance (Eckerli and Osterrieder, 2021), arts, video games, image and video restoration, etc. (Pavan Kumar and Jayagopal, 2021; Wang et al, 2021c).

However, training a GAN is not straightforward despite the numerous GAN architectures and training methods, as well as the diversity of CV-based industrial applications, image modalities, and challenges. Most authors focused on creating GAN taxonomies based on ML general training taxonomies (Chen and Jia, 2021), or GAN's architectures and loss functions (Wang et al, 2021c). From another perspective, some others concentrated on image generation or I2I exclusively (Pang et al, 2021; Chen and Jia, 2021; Wang et al, 2020c; Li et al, 2020a) in a general manner (Farajzadeh-Zanjani et al, 2022; Pavan Kumar and Jayagopal, 2021) or projected their reviews in particular fields (Cai et al, 2021; Yi et al, 2019; Eckerli and Osterrieder, 2021; Yinka-Banjo and Ugot, 2020; Koshino et al, 2021). On another side, the fourth industrial revolution, aka. Industry 4.0 aims to deploy autonomous robots to assist human workers in their daily tasks (Schuh et al, 2017; Naumann et al, 2023; Tang and Veelenturf, 2019). Despite the superiority of CV-based observations over human observations, CV-based application is not widely integrated into the industrial workflow (Rutinowski et al, 2022). Moreover, to the best of our knowledge, there is no dedicated GAN overview to support the industrial field. In this paper, we aim to fill this gap by providing a comprehensive review of various existing GANs and their applications in Industry 4.0: We compare existing methods by training on our rendered synthetic datasets covering different, simple, and complex 3D scenes composed of industrial assets. We generated paired images with multiple modalities for training various domain transfer architectures. We also applied image transformations such as kernel filters, color augmentation, cropping, etc., to imitate real industrial image settings and outputs for training and testing image-to-image (I2I) translation models.

This review is designed to provide a survey on the most notable GANs and a comprehensive implementation in industrial applications covering topics from dataset acquisition to appropriate GAN architecture selection and assessment of generated image quality. In Section 2, we provide an overview of GANs and their feasibility for industrial applications and information on acquiring appropriate training datasets. Then, in Section 3, we present our synthetic data generation pipeline and our rendered datasets used in our further experimentations: In Section 4, we delve into the latest techniques for image generation, including the generation of global and local features. Subsequently, in Section 5, we examine approaches for texture generation. Moreover, in Section 6, we compare different domain transfer I2I translations.

Furthermore, in Section 7, we focus on method-specific I2I applications such as image expansion, de-filtering, and super-resolution. Finally, we present various methods for assessing GANs and a taxonomy for common failure modes in Section 8, before concluding and sharing our thoughts in Sections 9 and 10.

2 GAN Applications in Industry 4.0

CV-based applications can be widely used in today's industry for tasks such as training robots, decision-making, and assisting human workers in increasing productivity. Among the available techniques, GANs have proven to be the most effective and commonly used architecture for producing high-quality images (Chen and Jia, 2021; Farajzadeh-Zanjani et al, 2022). GANs have also been widely adopted in a wide range of real-world applications. However, the literature presents limited GAN implementations in the industrial IoT (IIoT) and Industry 4.0 fields related to privacy and cyber-physical systems (Ashok et al, 2023; Hindistan and Yetkin, 2023; Nedeljković and Jakovljević, 2022), rare event simulation (Baldvinsson et al, 2022), resource orchestration (Gupta et al, 2023), defect and anomaly detections in image (Li et al, 2022; Bougaham et al, 2021) or sound (Hatanaka and Nishi, 2021) data, and document restoration (Sharma et al, 2019). In this review, we will elaborate more and focus on the capacity of GANs to fill the gap in current CV-based industrial applications based on the industrial applications and specific use cases.

2.1 GAN Motivation

Basically, GANs are designed to replicate the probability distribution of a dataset, to generate new, previously unseen data that follows the same distribution (Goodfellow et al, 2014). To achieve this, GANs must produce diverse images within each class and different across classes (Arora and Zhang, 2017). Specifically, considering C classes in the training set, an optimal GAN should satisfy inter-class and intra-class diversities (Arora and Zhang, 2017). Whereby, in an ideal case of a biased dataset, for n generated images, it produces $\frac{n}{C}$ images for each class. Moreover, each subset is representative of its initial class samples, making it more challenging to achieve (GM et al, 2021; Shahbazi et al, 2022).

2.1.1 Conditional vs Unconditional

Vanilla GAN: Technically, GAN is structured as a zero-sum game, i.e., minimax, between two neural networks: a generator G and a discriminator D. A generator network learn how to create artificial data that is realistic enough and similar to the training dataset, so it fools the discriminator. In its turn, D is responsible for distinguishing between real-world samples and artificially generated data samples. Consequentially, a GAN is successfully trained when both models are achieved (Goodfellow et al, 2014). However, because the loss functions may be stuck in a local minimum, leading to mode collapse failure, Arjovsky proposed replacing the minimax loss function with a Wasserstein loss function (WGAN) (Arjovsky et al, 2017). In this case, the discriminator does not categorize instances as real or fake, but it generates a number that the higher it is, the more realistic the instances are. This metric is not limited between 0 and 1. Therefore, it is not a threshold-based comparison.

Conditional GAN (cGAN): It is a version of GAN that applies conditional settings to both the generator and discriminator networks (Mirza and Osindero, 2014). These conditional settings can include auxiliary information such as class labels (Karras et al, 2020a), instance images (Casanova et al, 2021), or data pairing (Isola et al, 2017). The generator inputs both the latent space and class information condition and produces images. The conditionality applied to both networks is necessary for the generator to generate outputs that satisfy the discriminator (Boulahbal et al, 2021). Additionally, cGANs converge faster than classical GANs as the generated images follow a certain pattern.

2.1.2 Learning Overview

There are four major GAN learning methods based on how training datasets are handled: supervised, unsupervised, semi-supervised, few-shot, and transfer learning. Supervised learning (SL)

uses labeled datasets to train GANs, but acquiring labeled data can be time-consuming and have a likelihood of human error (IBM Cloud Education, 2020). Unsupervised learning (UL) does not use labels or domain pairings but requires a larger dataset for optimal performance (Chen and Jia, 2021). Semi-supervised learning (SSL) uses a dataset with few labels and aims to label the remaining images (Mustafa and Mantiuk, 2020). Few-shot learning (FSL) models are based on very few (Benaim and Wolf, 2018; Cohen and Wolf, 2019) or even a single image training dataset and are evaluated based on their performance (Park et al, 2020a; Lin et al, 2020; Shaham et al, 2019). Moreover, regarding small training datasets, Transfer learning (TL) is a method that leverages knowledge learned from one task and applies it to another related task (Pan and Yang, 2009). It is useful when data or resources are scarce in the target task, and it aims to improve the model's performance by transferring the knowledge from the source task. More details are available in Appendix C.

2.1.3 GAN Inversion

Reciprocally, GAN inversion is a process that aims to invert a given image back into the latent space of a pre-trained GAN model, such that the image can be faithfully reconstructed by the generator (Pasquini et al, 2023). This process allows for a better understanding of the GAN's internal representations and can enable applications such as image manipulation, restoration, interpolation, style transfer, and compressive sensing. Additionally, researchers are looking into the application of GAN inversion for dissecting GANs to understand their internal representations, so they figure out what GANs do not learn and to examine how realistic images can be generated by GANs (Xia et al, 2022).

2.2 Industrial GAN Applications

We can find and suggest GANs in a wide range of applications in various industrial sectors. Some of the notable examples include the following:

1. **Data Augmentation**: It is the primary goal of GANs in the industrial sector to train deep learning models, such as those used in robotics for object detection and manipulation, surveillance, and transportation tasks. Acquiring image datasets from inside factories can be challenging due to strict regulations and permissions aimed at ensuring privacy, security, and safety. The augmentation can affect the lack of specific assets by mimicking the distribution of another asset's rich dataset.
2. **Safety**: Uncommon and rare industrial scenarios such as incidents, collapses, or fires pose a high risk and are prohibited due to safety precautions. However, gathering the correct data at such moments is crucial for analytics, but RGB images may not be sufficient to fulfill most application requirements. Therefore, translation methods can replicate an image in another modality, such as thermal imaging or depth maps.
3. **Bridging Sim-to-Real Gap**: Creating realistic textures for 3D assets used in digital twining increases the virtual scene realism, and therefore increasing the performance of training robot in simulation, predicting and mitigating severe issues in industrial settings. Moreover, GANs can bridge the gap between simulation-based rendered images and real images by transferring between both domains.
4. **Resource Optimization**: Capturing data for large, heavy, or rare industrial assets can be expensive and time-consuming, especially when moving or borrowing such assets. GANs can disentangle existing image features and alter the environment with additional randomizations of different setups.
5. **Robot Fine-Tune**: GANs can be used to optimize robots' decision-making and increase their CV model prediction by translating a training dataset domain into the robot's camera sensor domain before training and deploying a personalized model, or reciprocally translating the captured images into the dataset domain before inference.
6. **Data Reusability**: Existing and previously collected data is "precious" and valuable data (Sharma et al, 2019). Therefore, GAN restores corrupted and degraded images suffering from noise, poor resolution, lens distortion, and low brightness or saturation,
7. **Privacy**: GAN can be used to omit/replace confidential data which are part of real images,

such as human faces, serial and ID numbers, operation barcodes, etc.

8. **Authenticity Check**: GAN inversion is used to verify the authenticity of ideas and designs. The method involves training a GAN model on a set of existing product images or sample designs and then using it to compare new designs with the original images. The process starts with identifying the point in the pre-trained GAN's latent space that closely reconstructs the new design, then projecting it back into the image space and comparing it with the original image. This approach allows for distinguishing between authentic and fake designs with high accuracy. Furthermore, it can be used to monitor and ensure the quality of products during the production process.

3 Experimental Setup

Data Description: In this review, we will use in-house generated, synthetic data, which is part of SORDI.ai (Abou Akar et al, 2022), to evaluate and compare GANs in the context of Industry 4.0. These datasets were designed to showcase the strengths and limitations of the GAN architectures under examination. Instead of relying on commonly used datasets such as MS COCO (Lin et al, 2014), ImageNet (Deng et al, 2009), MNIST (Deng, 2012), Cityscapes (Cordts et al, 2015), GTA5 (Richter et al, 2016), etc. - that have been used in the original research papers - we benchmark the SOTA GAN performance on novel datasets generated through techniques such as domain randomization (DR) to create custom datasets such as low variation and high variation datasets, as well as paired datasets for image translation applications, e.g., supervised I2I translation. We used Unity and NVIDIA Omniverse as simulation software and NVIDIA RTX A6000 GPUs for rendering.

Data Acquisition: To render our dataset, we followed the following six steps:

1. To begin, we created various small environments with different ground and wall colors, textures, and lighting conditions.
2. We selected key industrial assets, such as: small load carrier (KLT) box, trolley, forklift, jack, pallet, stillage, smart transport robot (STR), and electrical jack.
3. These assets were then arranged logically and realistically to replicate industrial settings.
4. We applied DR affecting the asset positions, rotations, visibility, and the room components' textures and lights. The target of our DR is to control the dataset variability and complexity only.
5. We rendered images from these scenes from different angles and perspectives while always focusing on the main industrial assets. With the help of Isaac Sim, this simulation allows rendering images with multiple annotations/modalities, e.g., plain color, instance segmentation, semantic segmentation, depth, etc.
6. We removed defective and high-occluded images during a data cleaning phase.

In Appendix A, we present samples and details of the datasets. For more technical details, we recommend reviewing our previous work concerning synthetic data generation for industries (Abou Akar et al, 2022; SORDI.ai, 2023).

Experimental Material: We will delve into the various GAN architectures and their applications in the industry in the following sections. We will begin by discussing the most recent and significant GANs for image generation, focusing on global and local features and texture generation. Then, we will look into the various ways to apply GANs in the industrial field, such as domain transfer techniques for I2I translations and image restoration. Our evaluation is based on the literature review and qualitative results of some of the latest GAN architectures that we implemented and tested. We used NVIDIA RTX 3090 and NVIDIA Tesla V100 GPUs to train our GAN models using the previously mentioned datasets for all our experiments.

4 GANs for Image Generation

Image Generation is undoubtedly a key application of GANs, offering not only a new approach for data augmentation, but a revolutionary and limitless technique for expanding existing image datasets (Shorten and Khoshgoftaar, 2019). In this section, we distinguish between global and fine-grained features image generation. However,

in the global image generation technics, we distinguish between different GAN architectures that handle large-scale and small size training datasets. However, for the fine-grained features GANs, we focus on image disentanglement technics, and Text-to-Image synthesis approaches.

4.1 Global Image Generation

Global image generation is one of GAN's main tasks for data augmentation in case of data scarcity, especially since DL models, e.g., object classification and recognition, are "data hungry". As unconditional image generation is beneficial for single-class dataset augmentation, conditional image generation is more suitable for multi-class dataset augmentation. Afterward, we utilize knowledge transfer approaches such as transfer learning or domain adaptation to train models on the synthesized data and then to fine-tune the models on real data, or mixing both datasets in the training phase (Ravuri and Vinyals, 2019b).

4.1.1 Large Scale GAN Training

StyleGAN Retrospective: Progressive GAN, also known as **ProGAN** (Karras et al, 2017; tkarras, 2017b,a), is capable of generating high-resolution images by gradually increasing the resolution of the images during the training process. This approach starts by training on small-dimensional images and gradually increases the resolution to include fine details. This method is adopted by both the generator and discriminator networks, resulting in faster training times. ProGAN is considered the primary foundation of StyleGAN architecture.
Limitations: However, it is worth noting that despite its efficient nature, progressive growth-based GAN is known to generate phase artifacts (Karras et al, 2019) as shown in Figure 1.

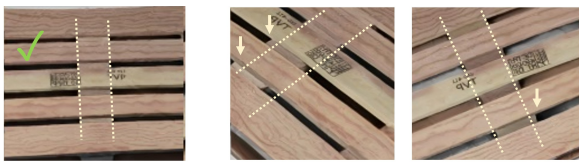


Fig. 1 (a) Good match (b) Mismatch of sub-wood plates in a generated euro pallet image

Inspired by the style transfer (Huang and Belongie, 2017; Jing et al, 2019), **StyleGAN** (Karras et al, 2019; NVLabs, 2021a) is an advanced version of the traditional ProGAN architecture (Karras et al, 2017) that generates high-resolution images with detailed style-level stochastic variations by using progressive resolution blocks, Gaussian noise injection, and AdaIN normalization (Huang and Belongie, 2017; Dumoulin et al, 2016, 2018; Ghiasi et al, 2017). However, it has drawbacks, such as water droplet artifacts and phase artifacts caused by progressive growing techniques. To overcome these limitations, **StyleGAN2** (Karras et al, 2020b; NVLabs, 2021b) was proposed as a revised version of StyleGAN. To reduce the artifacts, it replaces AdaIN normalization with estimated statistics, generates mipmaps (Williams, 1983) with a modified version of Multi-Scale Gradients for GAN (MSG-GAN) (Karnewar and Wang, 2020), and uses skip connections (Ronneberger et al, 2015) and residual networks (Gulrajani et al, 2017; He et al, 2016; Miyato et al, 2018) for the generator and discriminator, respectively. As a result, StyleGAN2 improves the training performance by 40% compared to StyleGAN, but still requires a large number of varying datasets to avoid discriminator overfitting and training divergence. More details are explained in Appendix D.

BigGAN: In 2018, Brock *et al.* published a new leveraged GAN architecture named BigGAN (Brock et al, 2018; ajbrock, 2019), which focuses on scaling up GAN models for class-conditional image generation. The architecture of BigGAN supports larger model parameters, such as an increased number of feature maps, larger batch sizes of up to 2048 images, and additional architectural changes such as skip connections and the truncation trick[1] to improve image quality.
Limitations: However, it is limited to generating images of 512x512 pixels resolution. Additionally, the generators of BigGAN are vulnerable to class leakage, local artifacts such as the checkerboard artifact (as shown in Figure 3), and collapse mode failures as reported in (Brock et al, 2018; Vo et al, 2022; Brownlee, 2019a). Furthermore, when

[1]The BigGAN truncation tricks consist of using different distributions for the latent space while training and inferring the generator.

we applied the IC-GAN (as described in IC-GAN in Section 4.1.1) for instance-conditional image generation using BigGAN, the results displayed texture blobs (as shown in Figures 4 and 2) due to the limited number of images in the training dataset that was similar to the condition image or that satisfied the conditional input, specifically in terms of the camera field of view and angle: in our case, ground-level side image of the trolly and the STR.

On the other hand, Sauer *et al.* proposed in (Sauer et al, 2022) the StyleGAN-XL model, a state-of-the-art (SOTA) approach for high-resolution image synthesis on large unstructured datasets such as ImageNet. By training the model using the Projected GAN paradigm, neural network priors, and a progressive growing strategy, they improved the performance of the latest StyleGAN3 generator, which performs poorly on large unstructured datasets such as ImageNet. StyleGAN-XL achieved a new SOTA in large-scale image synthesis, with the ability to generate images at a resolution of 1024x1024 pixels. Additionally, the model can invert and edit images beyond the narrow domain of portraits or specific object classes.

Limitations: One limitation of the proposed StyleGAN-XL model is the increased computational cost due to its larger size, three times larger than StyleGAN3. Future research could explore GAN distillation methods that balance performance and model size. Additionally, StyleGAN-XL is based on StyleGAN3. Therefore, it inherits the reduced semantic controllability introduced to achieve equivariance. Furthermore, while the model could benefit from being tested on larger and more diverse datasets, such datasets are currently unavailable.

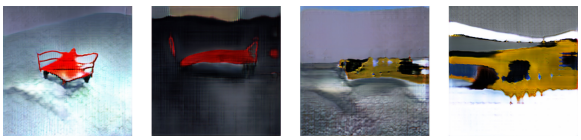


Fig. 2 BigGAN image generation with texture blob failure - incorrect object anatony failure

IC-GAN: As previously stated, traditional cGANs require a significant amount of labeled data, which may be infeasible to acquire. To address this issue, a team at Facebook AI Research proposed in 2021 an extension of GANs known as Instance-Conditioned GAN (IC-GAN) (Casanova et al, 2021; Facebook Research, 2021). This method utilizes a feature cluster-based approach (Likas et al, 2003; Soucy and Mineau, 2001) to generate images using unlabeled data.

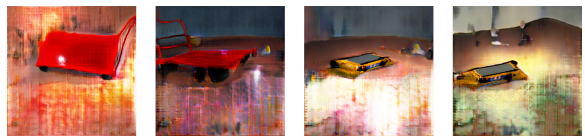


Fig. 3 BigGAN image generation with local checkerboard artifacts

During training, the discriminator considers each sample's latent vector neighborhood, which forces the generator to generate images similar to the sample's neighbors based on cosine similarities. This approach has been shown to be more effective than partitioning the data into clusters, particularly when generating images in an overlapped latent space. This is because, in such cases, the image may belong to two clusters, and their disjoint vectors may be vastly different. Once the model is trained, a single new image, or "instance," is sufficient to generate similar images to its closest neighbors in the dataset.

The Facebook team published a pre-trained network based on ImageNet. Still, it is inadequate for covering specific industrial assets, as demonstrated in Figure 5. As a result, we trained our network using BigGAN[2] using the same synthetic dataset as in previous studies. We noted that the generated images are similar not only to the main actor-instance but also to its scale, viewport, environment background, lighting, etc., as demonstrated in Figures 4 and I7. In contrast to traditional cGANs, which generate a diverse range of images containing the class label object in different viewports, backgrounds, etc., IC-GAN generates more specific, limited images that are similar to the instance neighborhood in the dataset.

Furthermore, Casanova *et al.* proposed a class-conditional IC-GAN (ccIC-GAN) for labeled data, in which the instance and class label conditions are combined. This allows ccIC-GAN to create

[2]IC-GAN supports two training backbones: BigGAN and StyleGAN2-ADA.

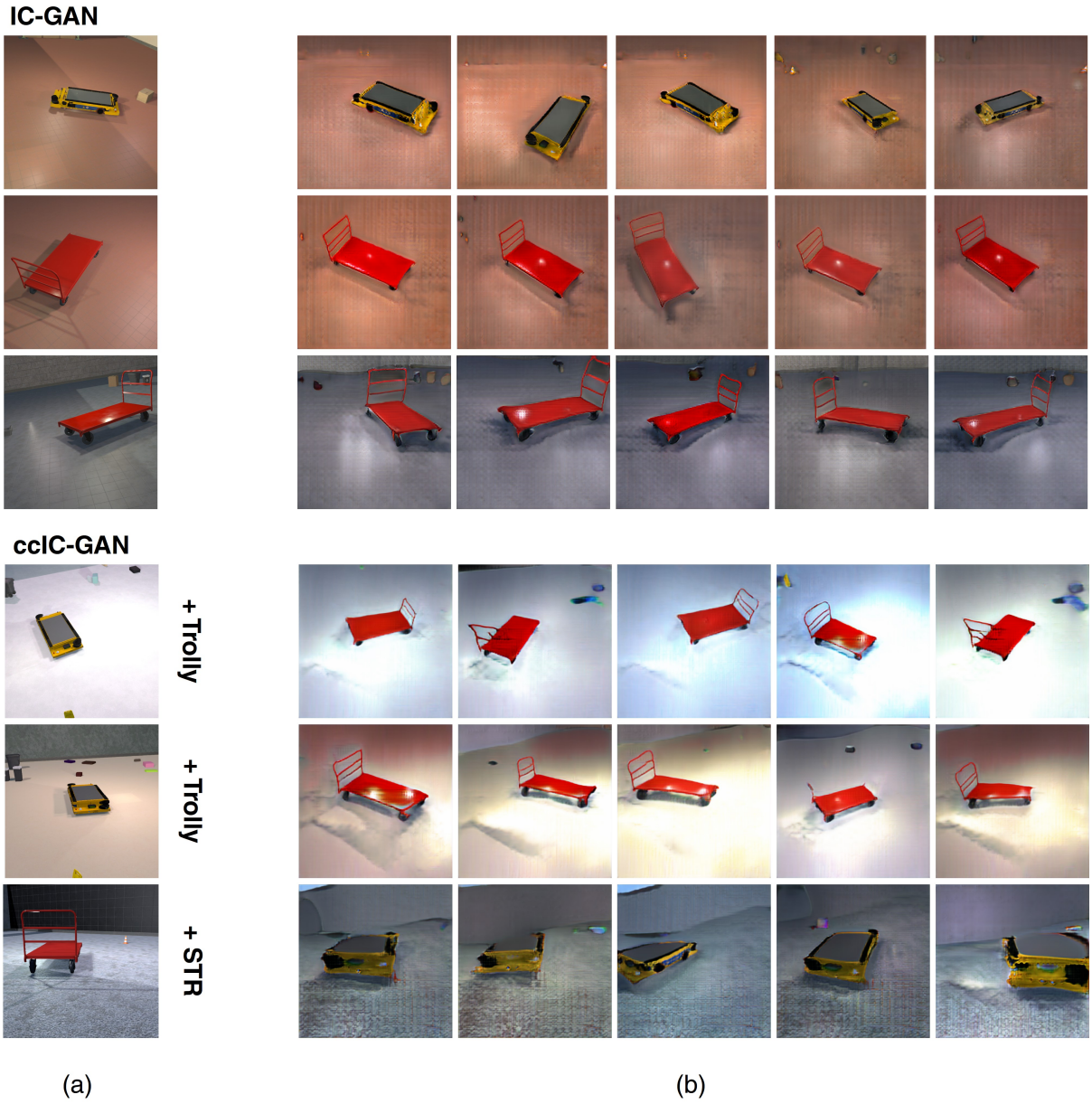


Fig. 4 IC-GAN and ccIC-GAN model trained from scratch: (a) Conditional instance (b) Generated output

the conditional class asset in the instance environment. For example, in Figures 4 and I8, we were able to generate images for the class label within the same surrounding environment of the image instance, taking into account the initial camera field of view, perspective, and distance from the target asset. However, some results appear "blobby," particularly concerning low-level ground angles, due to the lack of training data containing images captured at low-level ground angles. Additionally, while generating the class asset in the instance environment can result in more realistic and rare images, some results may be considered unusual (Meta AI, 2021). ccIC-GAN addresses the bias that objects may not be present in specific environments, for example, a forklift inside an office. More examples are shown in Appendix I.

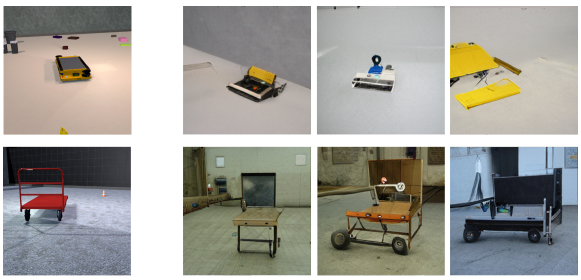


Fig. 5 ImageNet-based pre-trained model: (a) Conditional instance (b) Generated output

4.1.2 Limited Datasets GAN Training

Small dataset hampers machine learning training (Wang et al, 2020b), including generative networks. To tackle this problem, transfer learning (Pan and Yang, 2009; Weiss et al, 2016; Torrey and Shavlik, 2010) uses pre-trained models[3] to transfer knowledge and fine-tunes the model with fewer samples of data as in (Noguchi and Harada, 2019; Abbas et al, 2021; Wang et al, 2020a, 2018c; Mo et al, 2020), instead of training from scratch using random initializations and extensive data. For example, training from scratch takes around 4000 iterations, while approx. 300 iterations are enough in transfer learning. In Figure 6, we trained the first model to generate red trollies; then, we used the model's weights to initialize a model training for STR generation. Afterward, it took around 300 iterations and a small dataset of 10,000 images to train a GAN to generate images for an STR in a docking state or multiple instance generations as in Figure 6 c. and d. respectively. But, Karras *et al.* argues that the progress reverts as soon as reasonable FID is achieved.
Mo *et. al* agrees in (Mo et al, 2020) that GANs transfer learning is prone to overfitting or limited to learning small distribution shifts. Instead, they suggested FreezeD: It surpasses the literature by freezing the discriminator's lower layers and fine-tuning its upper layers. Additionally, many authors suggested data augmentation technics at the discriminator level for efficient GAN training (Zhao et al, 2020d,b; Tran et al, 2021). For instance, Bora *et al.* proposed in (Bora et al, 2018) AmbientGAN. They trained the discriminator not in the raw data domain but in the measurement domain. The measurement domain contains noisy, blurred, corrupted, and missing data, i.e., occlusion augmentation. The authors proved that it is still possible to recover the true/good distribution by training the generator if the measurement process is invertible under a certain probability. This concept was highly recommended to train GAN when it is impossible to obtain fully-observed image samples but some partial and noisy samples.

StyleGAN2-ADA: The NVIDIA team in (Karras et al, 2020a) made modifications to the discriminator network of StyleGAN2 by applying various augmentations such as pixel blitting, geometric and color transforms, filtering, additive noise, and cutout to each image presented to the discriminator. All augmentations were applied with the same probability unless they were skipped. The probability threshold of non-leaking augmentations was determined to prevent unwanted stochastic augmentations during generation. The probability parameter is not fixed and is highly dependent on the sensitivity, properties, and size of the dataset[4], as well as the training setup. To address these limitations, Karras *et al.* introduced an adaptive control scheme (StyleGAN2-ADA) (NVLabs, 2020). Later, NVIDIA Labs published a PyTorch implementation of StyleGAN2-ADA (NVLabs, 2021) that is 5-30% and 35% faster in training using NVIDIA Tesla V100 GPU and high-resolution inference, respectively, when compared to the previous TensorFlow implementation (NVLabs, 2020).
Additionally, The use of a conditional GAN is intended to enable class-based image generation, as depicted in Figures 7 and I6.

Limitations: Upon visual inspection of the generated images, it was observed that all details were "glued" to specific image coordinates instead of the appropriate parent object surface (Karras et al, 2021). This resulted in negative images and signal effects on the generated output, such as per-pixel noise inputs, positional encodings, and aliasing (Azulay and Weiss, 2018; Zhang, 2019; Parmar et al, 2021).
In an additional experiment, an unconditional StyleGAN2-ADA model was trained using 20,000

[3]Pre-trained models are models that have already been trained on some other datasets.

[4]The larger the dataset size, the more harmful the augmentation is (Karras et al, 2020a).

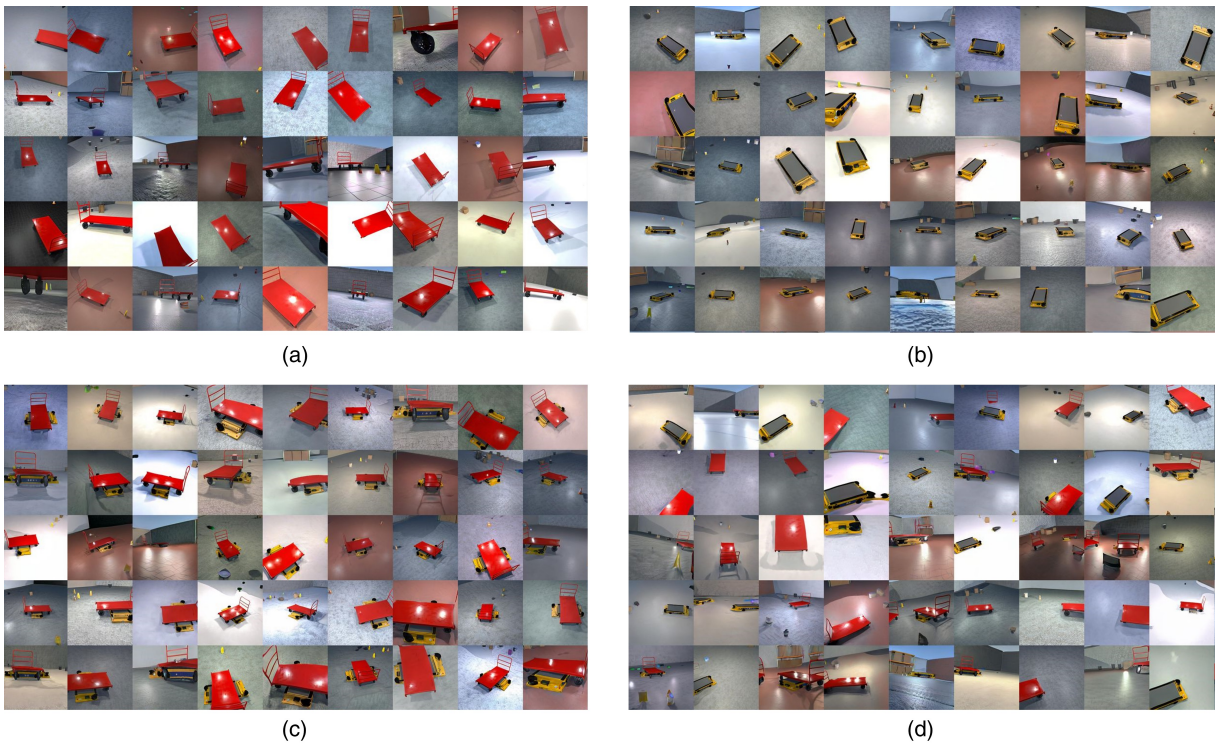


Fig. 6 GAN trainings from **scratch** (a) Red trolly @ 3898 iterations and **iterative transfer learning** (b) Smart transport robot (STR) @ 411 iterations (c) STR under a trolly @ 313 iterations (d) Mixed assets between trollies and STRs @ 296 iterations

training images of pallets and KLT boxes, with 10,000 images for each asset. It was observed that there were class leakage failures at iterations 4000 and 4200 (refer to Figure 8), such as (1) the presence of both assets, (2) collages of asset parts forming an unusual shape, (3) feature leakage from one class to another, for example, a pink KLT box with a wooden pallet's texture, or reciprocal, a wooden pallet with pink gradients.

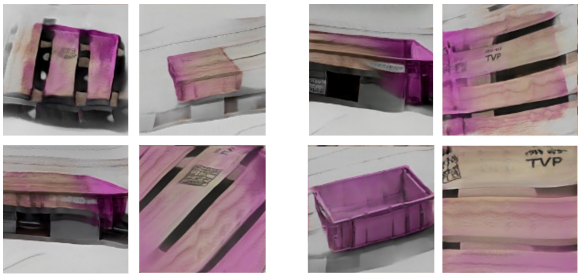
StyleGAN3 In Figure 9, it is evident that StyleGAN2-ADA's trolly texture sticks to the empty area between its hand bars, unlike in StyleGAN3's trolly, where this issue is greatly reduced, and we can see continuous light gradient effects. This issue, referred to as the "texture-sticking" problem, was addressed in StyleGAN3 (Karras et al, 2021; NVLabs, 2021) by incorporating a natural transformation hierarchy in the generator architecture of the GAN. The authors approached the problem by interpreting all network signals as continuous signals rather than discrete values. To achieve this, they adopted a classical Shannon-Nyquist signal processing framework (Shannon, 1949) in the continuous domain, using high-quality filters with over 100dB attenuation for antialiasing (as seen in Figure 10). Additionally, they employed Fourier features (Suzuki et al, 2018; Tancik et al, 2020), filtering, and 1×1 convolution kernels to make the generator equivariant to geometric transformations such as translation and rotation. Now, each sub-pixel inherited its position from the underlying coarse features. This results in a more natural motion (Alaluf et al, 2022) and enables the efficient generation of high-quality and realistic videos and animations.
However, as previously noticed in StyleGAN, a discriminator is responsible for multiple dependent randomizations between the local and global features. Thus, making the discriminator equivariant is worth investigating as well.

Mode collapse: Additionally, we observed that when trained on a dataset of 20,000 images of pallets and KLT boxes with low variation, the conditional training collapsed and generated distinct images for each class. These images still retained

(a)

(b)

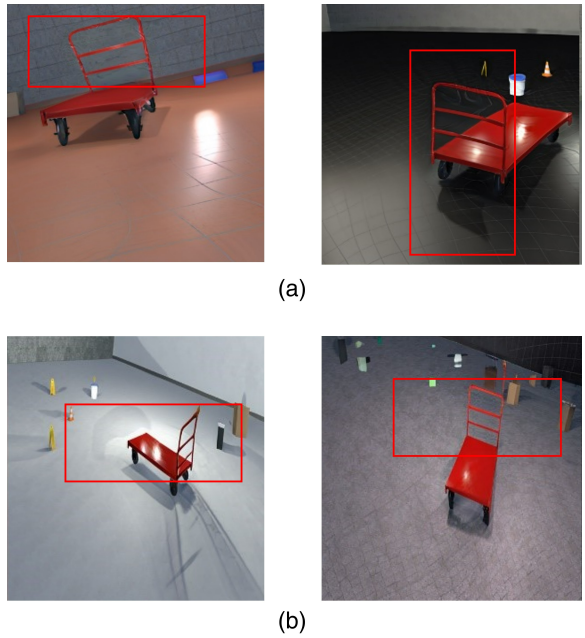
Fig. 7 (a) Conditional (b) Unconditional StyleGAN2-ADA generation sample



(a) (b)

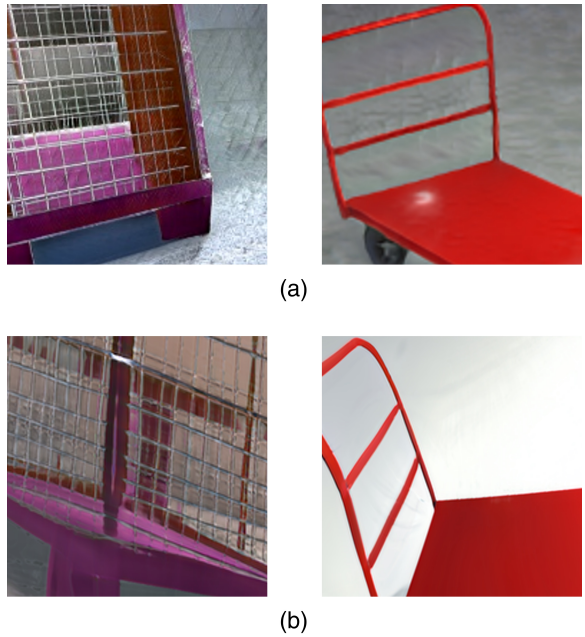
Fig. 8 StyleGAN2-ada class leakage failure at iteration (a) 4000 (b) 4200

a distorted shape or color of the initial class features (as shown in Figure 11 b.). On the other hand, unconditional training generated only a single "blob" image that lacked any recognizable characteristics of either class (as shown in Figure 11 a.). Additionally, we found that by introducing variation in only one class of images, it was possible to prevent the training from collapsing, even if the other class represented a lack of variation (as shown in Figure 12).



(a) (b)

Fig. 9 (a) StyleGAN2-ADA (b) StyleGAN3 generated trolly



(a) (b)

Fig. 10 (a) StyleGAN2-ADA (b) StyleGAN3 generated stillage (left) and trolly hand (right)

Training divergence: Additionally, it is important to note that the GAN training process highly depends on the quality and variation of the dataset used. In this case, it was observed that

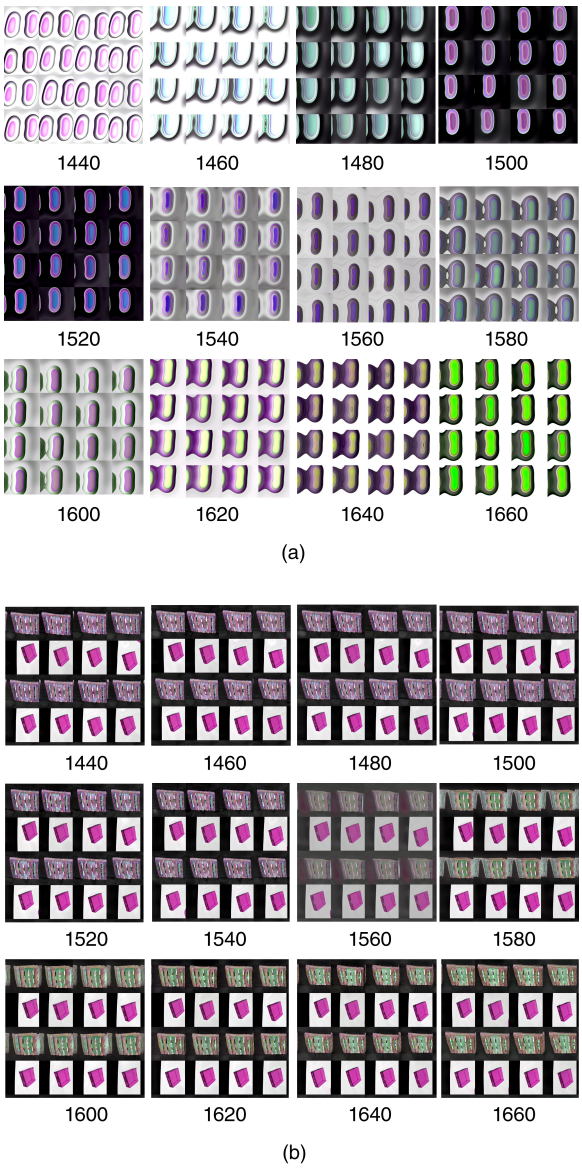


Fig. 11 Inter-class and intra-class collapse mode with (a) unconditional (b) conditional training styleGAN3 respectively

the previous training - referring to Figure 12 c. - diverged and generated only black images for both classes after a batch of good generation iterations, as shown in Figure 13. This phenomenon, known as overtraining, can negatively impact the quality of the generated images. To address this issue, Mittal (Mittal, 2019) suggests monitoring the Memorization-informed FID (MiFID) and implementing an Early Stopping technique to achieve the best results. It is also important

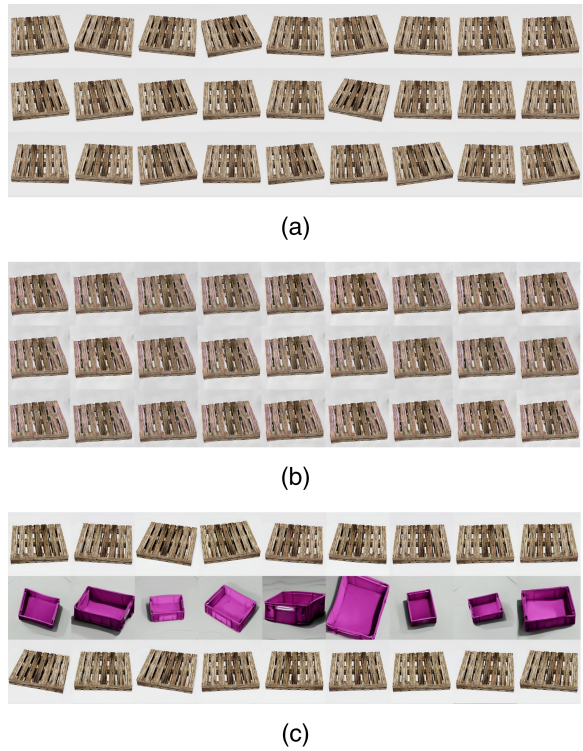


Fig. 12 (a) Pallet only training dataset (b) StyleGAN3 pallet generation - collapse mode (c) StyleGAN3 pallet with a variation of KLT Box generation

to note that the GAN training process typically includes a period of stability, during which the highest quality images are produced (Brownlee, 2019b). However, a quantitative-based "Early Stopping" method may not always be sufficient to detect this period, as the losses may fluctuate randomly without indicating any issues (Pasini, 2019). Therefore, keeping the training running longer and then, using opinion-based selection for the GAN checkpoint is recommended.

4.2 Fine-Grained Image Generation

Previous GANs focused on coarse-grained image generation, where many features in a single image are considered (Karras et al, 2019). This approach simultaneously considers a significant amount of information, such as the asset type, background, position, light, shade, texture, etc. However, in many industrial applications, fine-grained object recognition (Zheng et al, 2019) is required, such as detecting small tools in a load carrier box, identifying brand logos, or determining 3D pose

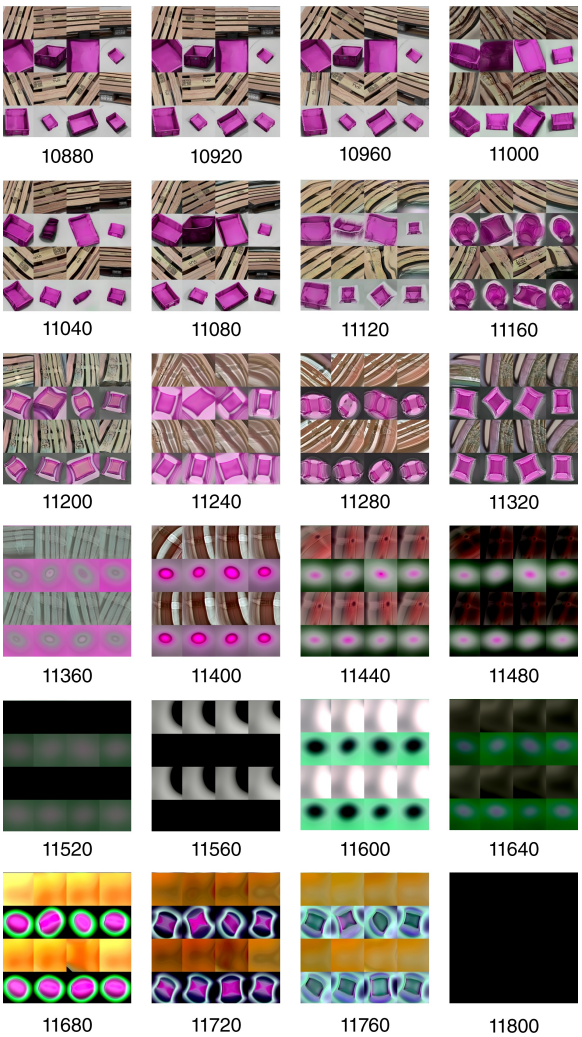


Fig. 13 Training divergence: Image generation sample at different training iterations from 10880 to 11800 with a step of 40

within a static 2D image. Still, fine-grained based applications face data scarcity limitations, such as having only a single studio image for each asset which can be scratched or having a limited number of images with partial views and a limited number of viewports[5] (Karlinsky et al, 2017). Conversely, applying coarse-grained-based GANs cannot distinguish intra-class subtle fine-grained features (Chen et al, 2022b), and therefore may easily collapse. To overcome these limitations, it is possible to augment the training dataset by either focusing on specific features and generating specific details variation, or interpreting, manipulating and controlling existing image features (Shen and Zhou, 2021; Härkönen et al, 2020; Roich et al, 2022; Pan et al, 2023). In this section, we present two approaches for fine-grained image synthesis by either disentangling features from existing images and then merging them in new images, or by prompting a significative text describing the output images.

[5]These steps are followed by geometric and photometric transformations to augment the dataset.

4.2.1 Features Dissentanglement

Existing GANs such as InfoGAN (Chen et al, 2016), FineGAN (Singh et al, 2019), and MixNMatch (Li et al, 2020c) can discover and disentangle common features from existing input latent vectors and apply them to the generated synthetic output (More details in Appendix E). For instance, based on FineGAN, MixNMatch disentangles in a 2-stage approach, four factors: background, object pose, object shape, and object texture, with minimal supervision. In Figure 15, we can see the combination of the KLT box texture, ground texture, and ground color features in the generated output.Moreover, aside from the hierarchical architectures, other researchers adopted VAEs (Bao et al, 2017; Luhman and Luhman, 2023), GAN inversion (Song et al, 2023; Liu et al, 2023), regularized the latent space's spatial organization (Chen et al, 2022b), or attention mechanisms (Wang et al, 2021a; Cai et al, 2019).

4.2.2 Text-to-Image Synthesis

Text-to-image synthesis is another recent and trendy approach to generating images with detailed descriptions. It is a task in which a model generates a visually realistic image based on a given text description; it allows for capturing the meaning of the text and converting it into an image. It has many applications, such as computer vision, natural language processing, and robotics. One approach they mention is the IC-GAN+CLIP (Radford et al, 2021; openai, 2022; Open AI, 2022), which uses the Contrastive Language-Image Pre-Training (CLIP) method to generate images based on text descriptions. CLIP is trained on a large dataset of more than 400 million image-text pairs scraped from the internet with

Fig. 14 DALL.E 2 experimentation

text description[6] and can predict entire classes of images it has not seen before, making it a bridge between computer vision and natural language processing. However, the authors note that prompt engineering is necessary for better results for each class. Similarly, StyleCLIP (Patashnik et al, 2021) considers StyleGAN image generation using the CLIP to modify the latent input vector. StyleCLIP suffered to reach visually diverse datasets generation as in AFHQ wild and LSUN Church datasets.

Another approach discussed is two-stage architecture-based GANs to generate high-resolution, photo-realistic images from text descriptions, such as StackGAN (Zhang et al, 2017a), StackGAN++ (Zhang et al, 2018a) and AttnGAN (Xu et al, 2018). This architecture is inspired by the process of a painter creating a painting, with the first stage creating a low-resolution image and the second stage refining it. The authors note that this architecture mainly analyzes complex text conditional descriptions to generate the main target object, e.g., a bird with a yellow crown and a black eye ring, a red bird with a white and very short beak, etc. Still, much research is needed in the area of complex-wide scenes with multiple objects.

[6] WIT400M dataset (Radford et al, 2021)

In addition, authors in (Ye et al, 2021) propose a contrastive learning approach to improve synthetic images' quality and semantic consistency in text-to-image synthesis. They evaluate their approach on two popular text-to-image synthesis models, AttnGAN and DM-GAN (Zhu et al, 2019) using COCO and CUB datasets, and find that it significantly improves the quality of synthetic images as measured by IS, FID, and R-precision.
Another work (Tao et al, 2022) proposes Deep Fusion Generative Adversarial Networks (DF-GAN) that are simpler but more effective for synthesizing realistic and text-matching images. They proposed a novel one-stage text-to-image backbone that directly synthesizes high-resolution images without entanglements between different generators, a novel Target-Aware Discriminator composed of MatchingAware Gradient Penalty and One-Way Output, which enhances the text-image semantic consistency without introducing extra networks, a novel deep text-image fusion block, which deepens the fusion process to make a full fusion between text and visual features. The approach followed previous work experimentation strategies and was also tested on COCO and CUB datasets.
Huang *et al.* introduced in (Huang et al, 2022) the Product-of-Experts Generative Adversarial Networks (PoE-GAN) framework, which can synthesize images conditioned on multiple input modalities or any subset of them, even the empty set. It learns to synthesize images, as in MM-CelebA-HQ and MS-COCO, with high quality and diversity and outperforms the best existing unimodal conditional image synthesis approaches when tested in the unimodal setting.
Finally, Open AI researchers launched a trendy and powerful transformer language model DALL.E 2 (Ramesh et al, 2022), a successor of DALL.E (Ramesh et al, 2021) with CLIP latent. It considers attributes, style, and concepts while interpreting the text prompt to generate photo-realistic images. However, DALL.E releases are out of the scope of our review because they are based on VAE and diffusion technologies, respectively.
Limitations: Still, image generation on CLIP struggles to binding attributes leading to mixed colors and realistic scale mismatches (check Figure 14 b, h), producing details in complex images as a big number of assets, or light conditions in Figure 14 i, e respectively, and starting to miss elements as in Figure 14 d, h, j. Plus, we found that it widely covers industrial and logistic assets (Figure 14 f). Instead, it produces high asset variations in a small synthesized sample, in terms of shape, size, color, etc. (Figure 14 a, c), but which may not answer targeted industrial applications as detecting a specific tool with predefined dimensions. Moreover, it did not translate number and count prompts conditions as in Figure 14 g.
However, in practice, the captions annotated by humans for the same image can have a large variance in terms of content and choice of words, which can lead to synthetic images deviating from the ground truth. This is due to the inability of the model to generalize to unseen data, as it has only been trained on a specific set of captions and images. This can be particularly problematic when working with datasets that are not diverse or are limited in size. Furthermore, a model trained on one dataset may not be able to generalize well to new datasets with different captions and images. This limitation is an important area of research in text-to-image synthesis, as it is crucial for the model to be able to generate high-quality images that are consistent with the given text description.
StyleGAN-T is a text-to-image synthesis model that aims to regain competitiveness for GANs compared to diffusion models. The model, proposed by Sauer *et al.* (Sauer et al, 2023), is based on the StyleGAN-XL architecture and is specifically designed to meet the requirements of large-scale text-to-image synthesis such as large capacity, stable training on diverse datasets, strong text alignment, and controllable variation vs. text alignment tradeoff. One key advantage of StyleGAN-T over diffusion models is its fast inference speed, as it only requires a single forward pass, while diffusion models require iterative evaluation to generate a single sample. Furthermore, StyleGAN-T also guarantees smooth latent space interpolation. The architecture was trained on a total of 250M text-image pairs from different public datasets, at a resolution of 64x64 [7], with

[7]Knwoing the ability of GANs to generate high-resolution images, the authors prioritize their budget cost instead of spending them on super-resolution stages. Still, it was less

a budget of 4 weeks on 64 NVIDIA A100s. As a result, StyleGAN-T outperformed current SOTA diffusion models at a resolution of 64x64.
Limitations: At 256x256, while StyleGAN-T improves upon the zero-shot FID previously achieved by a GAN by half, it still falls behind the SOTA diffusion models. Additionally, StyleGAN-T uses CLIP as part of the loss function training[8]. Therefore, it sometimes inherits the same limitations as previously mentioned above.

4.3 Summary

Table 1 provides an overview of GAN-based image generation techniques previously discussed in the literature. These techniques include those based on various StyleGAN models, transfer learning, conditional generation, training with both limited and large-scale datasets, as well as fine-grained feature disentanglement.

5 GANs for Texture Generation

While many texture classifications exist: highly random, semi-structured, or regularly repeated (Liu et al, 2002), non-parametric, or parametric (Jetchev et al, 2016), isotropic, or anisotropic (Shaham et al, 2019), artistic, shape, or natural (Chen et al, 2022a), stationary, or globally-variant (non-stationary) (Wei et al, 2009; Zhou et al, 2018), etc. textures and surfaces occupy a part of the industry, especially when detecting materials, scratches, or even augmenting data or implementing DR in the simulation to reduce the sim-to-real gap. SOTA includes efficient texture synthesis using DL and CNNs (Gatys et al, 2015a; Ulyanov et al, 2016a; Li et al, 2017; Liu et al, 2020). However, adopting the traditional GAN image generation architectures may collapse since the input images are similar and lack diversity. Although, randomly selected patches from the same texture images are perceived to be similar (Wei et al, 2009). Therefore, specific GANs can be considered for procedural texturing[9] (Portenier et al, 2020). Next, we will explore existing GANs for texture generation.

Spatial GANs: Inspired by DCGAN (Radford et al, 2015). Jetchev *et al.* introduced Spatial GAN (SGAN) as the first fully unsupervised texture synthesis method based on GAN (Jetchev et al, 2016). It allows real-time, fast, and scalable generation of high-resolution images and the ability to merge multiple source images to create new textures. Additionally, it supports seamless tiled texture[10] generation.
Limitations: However, due to the strong mixing characteristic of SGAN, it does not work efficiently on all texture classes such as non-mixing textures or statistical dependant patterns, e.g., chess grid pattern, aligned letters, etc. It will fuse all input together. Also, it does not support realistic texture morphing - a smooth transition between two or more textures.

Afterward, the authors improved SGAN and published a periodic SGAN (PSGAN) (Bergmann et al, 2017) to generate periodic textures and to blend between different textures creating new ones. Still, aperiodic textures are not supported, e.g., Penrose tiling, perspective projections, etc.
Limitations: Additionally, depending on Vanilla GAN, SGAN, and PSGAN inherit the same GAN problems as mode collapse, mode dropping, convergence problems, etc. (Bergmann et al, 2017; Alanov et al, 2019). Plus, the training is unsupervised, and therefore the generation is random, still similar to the training texture images, and cannot be controlled to specific textures class input or new unseen textures.

Implicit Periodic Field Network: Chen *et al.* argues in (Chen et al, 2022a) that visual pattern synthesis models are assessed based on the generated samples' authenticity, diversity, and scalability. To align with these three characteristics, the authors designed an Implicit

costly than Stable Diffusion cost (Rombach et al, 2022; CompVis, 2022)

[8] The use of strong CLIP guidance in the model can limit the diversity of the generated images and introduces image artifacts (Sauer et al, 2023)

[9] A procedural texture is when an algorithm generates the texture instead of relying on the time-consuming process of photogrammetry or error-prone projection of the texture mapping.

[10] A tiled texture is, when repeated side-by-side with a copy of itself, displays no visible seam or junction where the two tiles meet.

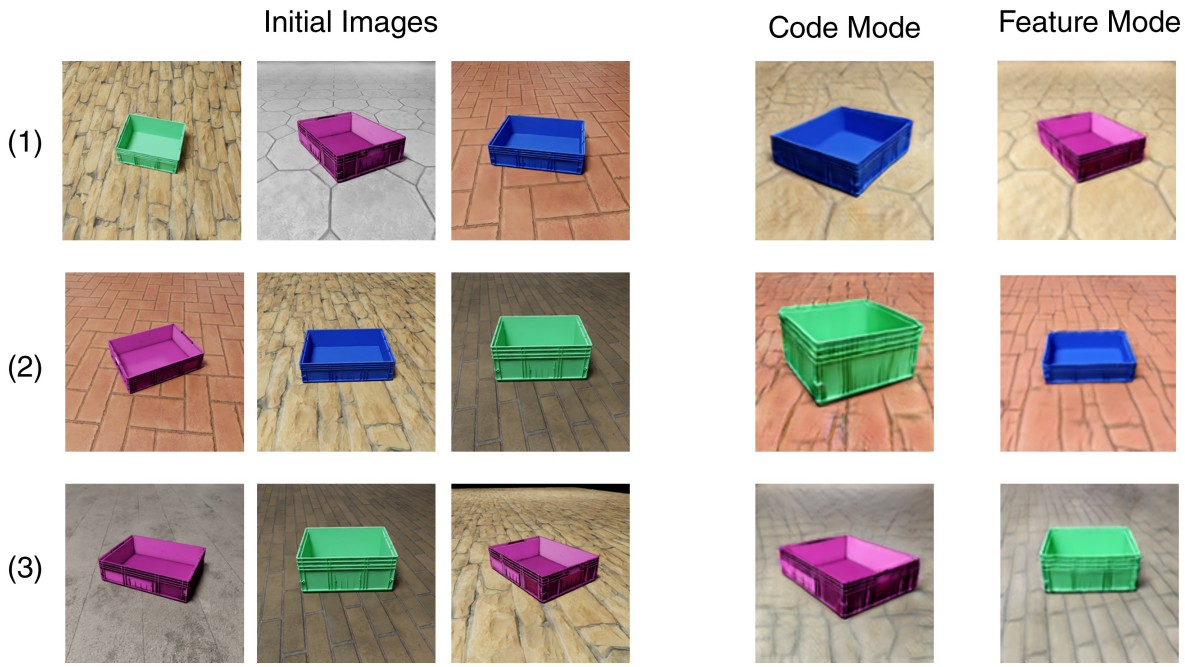


Fig. 15 MixNMatch with code and feature mode generation

Periodic Field Network (IPFN) as a combination of GAN and periodic encoding: It learns (1) high-frequency details using Fourier encoding (2) from different randomly shifted patches, and (3) periodicity for modeling the observed stationary variations from a single input training image.
Limitations: The authors developed their design because most natural patterns are stationary or bidirectional, without considering the synthesis of radial, web, or spiral textures.

LocoGAN: Struski *et al.* used local learning to train their convolutional GAN (LocoGAN) models so it fits on different image size datasets to generate infinite long images such as wallpapers and panoramic images (Struski et al, 2022). Applying LocoGAN to texture synthesis, the authors take a single high-resolution texture image and crop it to equal patches to create a new dataset out of it. Then the GAN generates periodic parts. Each fragment is taken and can be repeated many times.
Limitations: Despite the variety of the fragment itself, the whole synthesized texture sample is a set of repetitive "cloned" patterns; instead, it would be better if the fragment was expanded.

Multi-Texture Synthesis: Alanove *et al.* supports in (Alanov et al, 2019) multi-texture conditional generation and force cover all textures' latent representations in the training dataset by using a loss function that penalizes all wrong texture generations. Additionally, it can learn texture manifolds from high-resolution images.
Limitations: It requires a well data preprocessing and preparation to learn the manifolds. In addition, some texture manifolds could be unrealistic when they contain mixed patterns or features.

TileGAN: Früstück *et al.* were the first to attempt in (Frühstück et al, 2019) the problem of combining seamlessly multiple input images to generate a large-scale output image without boundary artifacts. Considering synthesizing an image with hundreds of megapixels, the proposed approach excelled quickly in producing large-scale maps from aerial images at different levels of detail.

Texture Mixer: Yu *et al.* trained in (Yu et al, 2019) an autoencoder and a cGAN to smoothly interpolate between different textures to synthesize a user-controllable and visually real-looking

Table 1 GAN-based image generation approaches (From top to bottom: general image generation, text-to-image synthesis, and fine-grained generation approaches

Architecture	Advantages	Drawbacks	Dataset
StyleGAN (Karras et al, 2019)	Generates similar image to the input training dataset	Produces droplet and phase artifacts	FFHQ, LSUN (Bedroom, Car, Cat)
StyleGAN2 (Karras et al, 2020b)	Fixes StyleGAN generation's artifacts	Necessities huge amount of training dataset as input	
Transfer Learning GAN (Karras et al, 2020a)	Necessities a small amount of training dataset	Overfits and limits the learning of small distribution shifts	FFHQ, LSUN (Car, Cat, Church, Horse)
StyleGAN2-ADA (Karras et al, 2020a)	Necessities only thousands of training data as input + Supports conditional-based generation	Produces per-pixel noise, positional encodings and aliasing	FFHQ, MetFaces, BreCaHAD, AFHQ (Cat, Dog, Wild), CIFAR-10
StyleGAN3 (Karras et al, 2021)	Solves StyleGAN2-ADA's "texture-sticking" and aliasing problems	Unlinks local features positions from their global features. Poorly performs on unstructured larger datasets as ImageNet	MetFaces-U, AFHQv2, FFHQ, Beaches[a]
BigGAN (Brock et al, 2018)	Supports class-conditional image generation with extra model parameters and larger batch sizes	Leaks class features from one class to another	ImageNet, JFT-300M[b]
StyleGAN-XL (Sauer et al, 2022)	Improved performance of StyleGAN3 on unstructured and large datasets. A modular framework that supports other GAN architectures	Models are three times larger than previous StyleGAN models. Semantic controllability is reduced for the sake of StyleGAN3's equivariance. Hence, it is hard to edit	ImageNet
IC-GAN (Casanova et al, 2021)	Generates images similar to the input image's neighborhood in the latent space	Depends on the architecture backbone's drawbacks: StyleGAN2-ADA or BigGAN	ImageNet, COCO-Stuff, Cityscapes, MetFaces, PACS, Sketches
ccIC-GAN (Casanova et al, 2021)	Desentangles the class conditioned object-of-interest in the image-conditioned environment	Generates low-quality images - sometimes unrealistic - if the object of interest's new pose is not highly occurrent in the training dataset	Same as previous
IC-GAN+CLIP (Meta AI, 2021)	Generates images based on text description in an environment similar to the image conditioned environment	Requires prompt engineering and does not satisfy a real industrial setting similar to our proposed environment	WIT400M
StyleGAN-T (Sauer et al, 2023)	Outperform diffusion models at 64x64 resolution images. Fast inference and smooth latent space interpolation.	It still falls behind diffusion models at 256x256 inference. Similar to DALL-E 2, STyleGAN-T inherits CLIP's limitations: attributes binding issues, scale mismatches, missing details in complex images, etc.	Union of: CC12m, CC, YFCC100m, Redcaps, LAION-aesthetic-6+
AttnGAN (Xu et al, 2018)	Generates text-conditioned images similar to a painting process (2 stages)	Focuses on modifying the object-of-interest, contrary to the static industrial assets, and not a whole complex multi-object scene	CUB, COCO
InfoGAN (Chen et al, 2016)	Disentangles common features between training dataset subsets for feature generation maximization	Varies only common features in the training dataset without considering multiple features from new multiple images	MNIST, SVHN house numbers, CelebA, 3D chairs
FineGAN (Singh et al, 2019)	Disentangles multiple features and combines them in a new background	Supports only latent code instead of images	CUB Birds, Stanford Dogs, Stanford Cars
MixNMatch (Li et al, 2020c)	Disentangles simultaneously, and with minimal supervision, four features from an image: background, pose, shape, and texture	Generates low-resolution images since it is based on FineGAN architecture	CUB Birds, Stanford Dogs, Stanford Cars

[a]Self-collected, proprietary, 20,155 512×512 beach images provided by Getty Images
[b]Internal Google dataset of 300M images dedicated for image classification tasks

single-output texture image. The interpolation task is executed after projecting the textures in the latent space.

Limitations: Training such a model requires a large dataset of textures highlighting rich intra-variability samples of each texture category.

Non-Stationary Texture Synthesis: Zhou *et al.* proposed in (Zhou et al, 2018) a new self-supervised GAN approach to expand by doubling the size of, mainly, non-stationary texture images while conserving and maintaining all visual characteristics and natural appearance similarities to the original exemplar. Stationary textures are

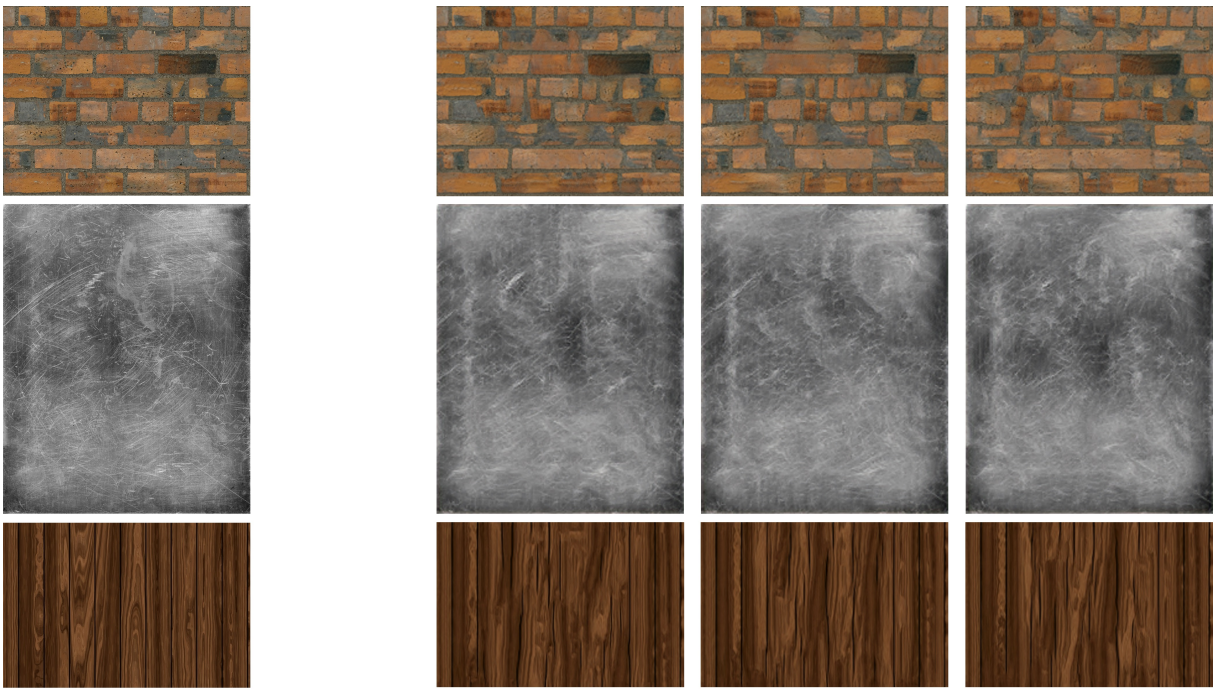


Fig. 16 SinGAN generation for (from top to bottom) brick, metal, and wood textures: (a) Input (b) Generated output

also supported, including regular, near-regular, and stochastic structures. The approach drops in the field of image translation application rather than the generation field since the GAN aims to expand a random cropped texture block belonging to the exemplar. Hence, the discriminator distinguishes between the produced extension vs the initial larger (doubled) block of the exemplar. Moreover, iteratively repeating the expansion cycle would lead to large-scale texture images.
Limitations: Existing models cannot generalize to new unseen data; hence, every new exemplar requires a dedicated generator to be trained. According to (Liu et al, 2020), the training phase is extremely slow - more than 6 hours using Tesla V100 GPU. However, the authors argue that the synthesis inference is extremely fast once the training is done.

SinGAN: SinGAN (Shaham et al, 2019) is an unconditional GAN trained on a single natural image. It consists of a pyramid of patch-GANs (Markovian discriminator) (Li and Wand, 2016; Isola et al, 2017) where training and inference are executed coarse-to-fine. The model collects patch distributions and trains at different scales of the complex image, capturing global properties such as shapes in the image and fine details such as texture information. Additionally, each GAN has small receptive fields and a limited capacity, preventing it from memorizing the whole single image, but however, can generalize well to unseen image inputs. Additionally, it is worth mentioning that Shaham *et al.* developed SinGAN to go beyond texture generation.
Limitations: Nevertheless, after the examination (check Figure 16) (tamarott, 2020; kligvasser, 2021), we found that only the image center differs from one generation to another. At the same time, it conserves the borders that are always similar to the initial image used for training. Liu *et al.* argue in their transposed convolution filter texture synthesis that SinGAN's inference is slow regarding a large number of textures (Liu et al, 2020). Yet, it reached competitive synthesis scores compared to other texture generation approaches, e.g., the previously mentioned non-stationary texture synthesis (Zhou et al, 2018). Although, Gu *et al.* proposed a new improved SinGAN with less

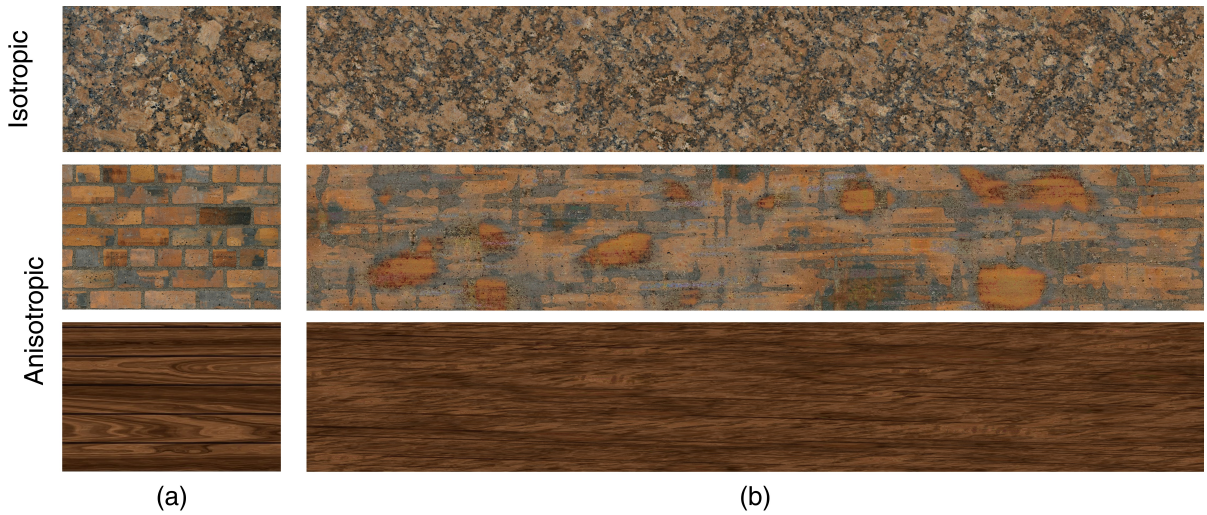


Fig. 17 GramGAN generation for (from top to bottom) granite (100,000 iterations), brick (139,000 iterations) and wood textures (145,000 iterations): (a) Input (b) Generated output

training time and an additional attention mechanism to increase image realism (Gu et al, 2021).

GramGAN: Unlike all the GANs mentioned above, GramGAN (Portenier et al, 2020) is a novel texture synthesis framework that aims to generate infinite and high-quality textures given a conditioned exemplar image of the target texture. From one side, despite the importance of the style loss function for anisotropic texture, e.g., wood, solely optimizing the style loss function causes artifacts as tiny replicas when zoomed-in (Mordvintsev et al, 2015). Conversely, the GAN loss is more effective on isotropic textures like granite. As a solution, Portenier *et al.* proposed a new loss function combining both style transfer (Gatys et al, 2015a) $\beta = 1$ and GANs (Goodfellow et al, 2014) $\alpha = 0.1$ to learn noise frequencies and to match Gram matrices so it generates highly realistic textures. A Gram matrix computes the style loss and extracts/captures the style of an image (Gatys et al, 2015b,a; Johnson et al, 2016; Gatys et al, 2016b; Ulyanov et al, 2016a).
While training the model or generating an image, it is essential to classify the texture as isotropic, e.g., stones, granites, etc., or anisotropic, e.g., wood, brick, etc.: the sample considers random slices while random rotations are restricted, respectively. Therefore, the parameters must be fine-tuned to satisfy a logical output (tportenier, 2020). However, in the proposed publication (Portenier et al, 2020), we found great interest in isotropic use cases over anisotropic considerations. In the anisotropic use case, we remark that the generated image shares common color distribution, gradients, and outlines with the training image. As displayed in Figure 17, the overall result for the brick and wood textures does not look the same as the training image, but both images can be semantically related.
Limitations: The training is time-consuming and necessitates many iterations (more than 100,000) for acceptable texture quality.

In Table 2, we summarize previously discussed GAN-based texture-generation approaches. It should be noted that some texture synthesis approaches may not generalize to new unseen data and require the training of a new dedicated GAN model, making it a time-consuming process. However, GramGAN effectively generates isotropic textures, while SinGAN is better suited for anisotropic textures. Additionally, while most texture synthesis methods may take a long time to train, they are relatively fast for a generation.

6 GANs for Domain Transfer

In industrial environments, multiple sensors with different functionalities and hardware configurations are often used in the same production area (Jacques and Christe, 2020), leading to various

Table 2 Texture Generation Approaches' Brief

Approach	Advantages	Drawbacks	Dataset
SGAN (Jetchev et al, 2016)	First fully unsupervised texture synthesis GAN in high resolution including tiles textures	Inefficient for all texture classes (only: stationary, ergodic and stochastic textures) and does not support texture morphing	Flower, *Amsterdam and Barcelona's Google Maps satellite views*[a]
PSGAN (Bergmann et al, 2017)	Supports periodic texture generation	Ineffiecient for aperiodic textures. Suffers from the same Vanilla GAN (dropping, collapse, convergence) problems	Oxford Describable Textures Dataset (DTD), Facades, *Sydney Google Maps satellite views*, *P6 and Merrigum House collected from furthermore Commons*
IPFN (Chen et al, 2022a)	Emphasizes local and stationary texture generation	Inefficient for radial pattern textures	DTD
LocoGAN (Struski et al, 2022)	Variety of generated texture fragments using single image high-resolution texture image	Final "infinite" image consists of cloned single-synthesized texture fragment	*self-developed dataset*
Multi-Texture Synthesis (Alanov et al, 2019)	Cover all dataset texture and generates texture manifolds from high-resolution images	Requires well-prepared and processed images before learning manifolds	DTD
TileGAN (Frühstück et al, 2019)	Fast generation large-scale images from multiple images at a different level of detail	Focusing on restoring large scale images, such as maps, by bringing additional textures and details to it - Training could last for days according to the authors' implementation	*self-collected datasets*[b]
Texture Mixer (Yu et al, 2019)	Smoothly interpolate different textures with high realism and user-controllability	Requires a rich intra-variability training dataset which is hard to find online	*self-collected animal and earth texture datasets*
Non-stationary Texture Synthesis (Zhou et al, 2018)	Expands by doubling non-stationary (and stationary) texture images while maintaining the natural appearance similarities - Once the model is trained, the inference is fast	Extremely time-consuming (more than 6 hours on Tesla V100 according to (Liu et al, 2020))	DTD
SinGAN (Shaham et al, 2019)	Able to generalize even with a single image input. Fast sampling (in terms of time, less than a second). Efficient for both: anisotropic and isotropic textures without producing repetitive global structures	Static fixed borders, and slow training (around 45-50 minutes on Tesla V100, or GTX 1060)	BSD100
GramGAN (Portenier et al, 2020)	Infinite width and height texture generation	Time-consuming, necessites a lot of training iterations. Mostly efficient for isotropic textures, e.g., stones, granite, etc.	*self-collected online stone textures dataset*

[a]Self-collected snapshots from Google Maps at specific GPS coordinates

[b]They collected terrain maps from Google Maps, satellite imagery from Landsat dataset (https://landsat.gsfc.nasa.gov/data/), and thousands of tile samples from high-resolution art and sky images.

image qualities, resolutions, color ranges, settings, and modalities[11]. This makes it difficult to train adaptive models for each robot, as collecting and labeling training datasets from different hardware is time-consuming, costly, and repetitive. As a solution, it is possible to transfer a perfectly collected dataset from its original domain to another application domain before training the model. Additionally, each sensor modality introduces a different type of information, such as segmentation images for autonomous driving tasks (Kaymak and Uçar, 2019; Geyer et al, 2020), depth images for depth estimation and 3D reconstruction tasks (Song et al, 2017), or thermal images for visualizing thermal distribution and outdoor security surveillance (Hasan et al, 2019). Therefore, multimodality fusion (Maqsood and Javed, 2020; Zhang et al, 2018b) aims to collect multi-source information for the same captured instance, extend knowledge, and deeply understand the captured

[11]A modality represents information in a specific medium (Bernsen, 2008; Tzovaras, 2008)

scene. However, capturing a single image with different modalities simultaneously requires different types of synchronized hardware or advanced image processing and AI models, which can be expensive, time-consuming, and confusing (Chen and Jia, 2021; Petrovic and Cootes, 2006). As a solution, Image-to-Image (I2I) translations can map different domains and transfer one image from one domain, such as modality, into another (Pang et al, 2021). This allows for a domain transfer that can translate images captured by different cameras to a single reference and trusted domain with better predictions, thus reducing additional training complexity. By using I2I translation and multimodality fusion, it is possible to overcome the challenges of variations in image quality and modality and to improve the performance of models trained on industrial data.

In further experimentation and as a proof of concept for some industrial Image-to-Image (I2I) translation applications, we focused on modern and widely supported I2I architectures. However, we distinguish between **two-domain I2I** and **multimodal I2I** translations. Pang *et al.* have divided the I2I methods into four categories in (Pang et al, 2021)[12]: Supervised, Unsupervised, Semi-Supervised, and Few-Shot trainings. These categories refer to the different levels of supervision provided during training. Supervised methods require paired data, unsupervised methods do not require any paired data, semi-supervised methods require some paired data, and few-shot methods require a small number of paired data.

6.1 Supervised Training

For a **single-modal output**, a source image is mapped to a target image, generating a single output. A robust supervised two-domain GAN architecture for single-modal output generation is Pix2Pix (Isola et al, 2017). Additionally, several variants have been proposed to address its limitations. For example, DRPAN includes a reviser to address Pix2Pix's blurry output. Wang *et al.* proposed Pix2PixHD (Wang et al, 2018a) to mitigate the instability and prone issues of the previous version while considering high-resolution images, and it supports image generation of up to 2048x1024 pixels. However, all versions of Pix2Pix fail to capture complex scenes, especially when both domains have severe deformation and evident different views. Other architectures, such as SelectionGAN (Tang et al, 2019), SPADE (Park et al, 2019), CoCosNet (Zhang et al, 2020b), CoCosNetv2 (Zhou et al, 2021), etc., have been developed to deal with cross-view translation problems, optimize the computational cost, or improve image quality. In technical terms, a single-modal

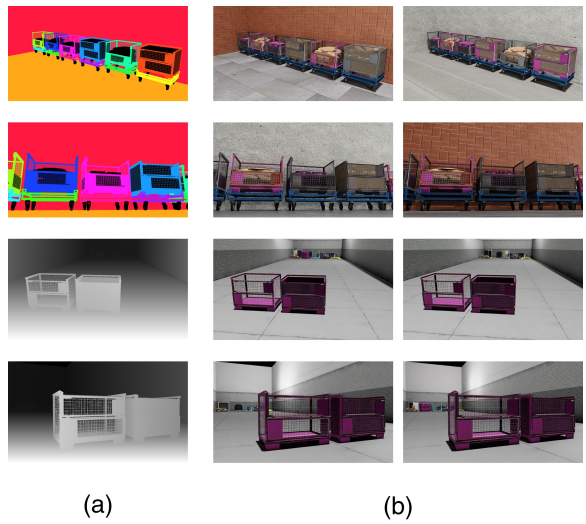


Fig. 18 (a) One to (b) many translations as ground truth paired data

output corresponds to a bijective function, that is, a one-to-one mapping (Chen and Jia, 2021). Precisely, a source image is mapped to a unique target image in a particular domain, such as translating a plain color image into an instance segmentation or depth image. However, the reverse translation is a non-injective, surjective function, resulting in multiple possible outputs, referred to as **multi-modal outputs**. For instance, as depicted in Figure 18, a depth image is a grayscale image with a single color channel, where each pixel has a single integer value ranging between 0 and 255. The higher the value, the darker the color, the greater the distance. Consequently, all asset pixels are mapped to black pixels, regardless of their specific attributes after a certain distance threshold. In industrial applications, translating a 2D plain color image into a depth image for depth estimation is considered more critical than reconstructing a plain

[12]For technical details, we recommend reading Pang *et al.*'s review of I2I methods and applications (Pang et al, 2021)

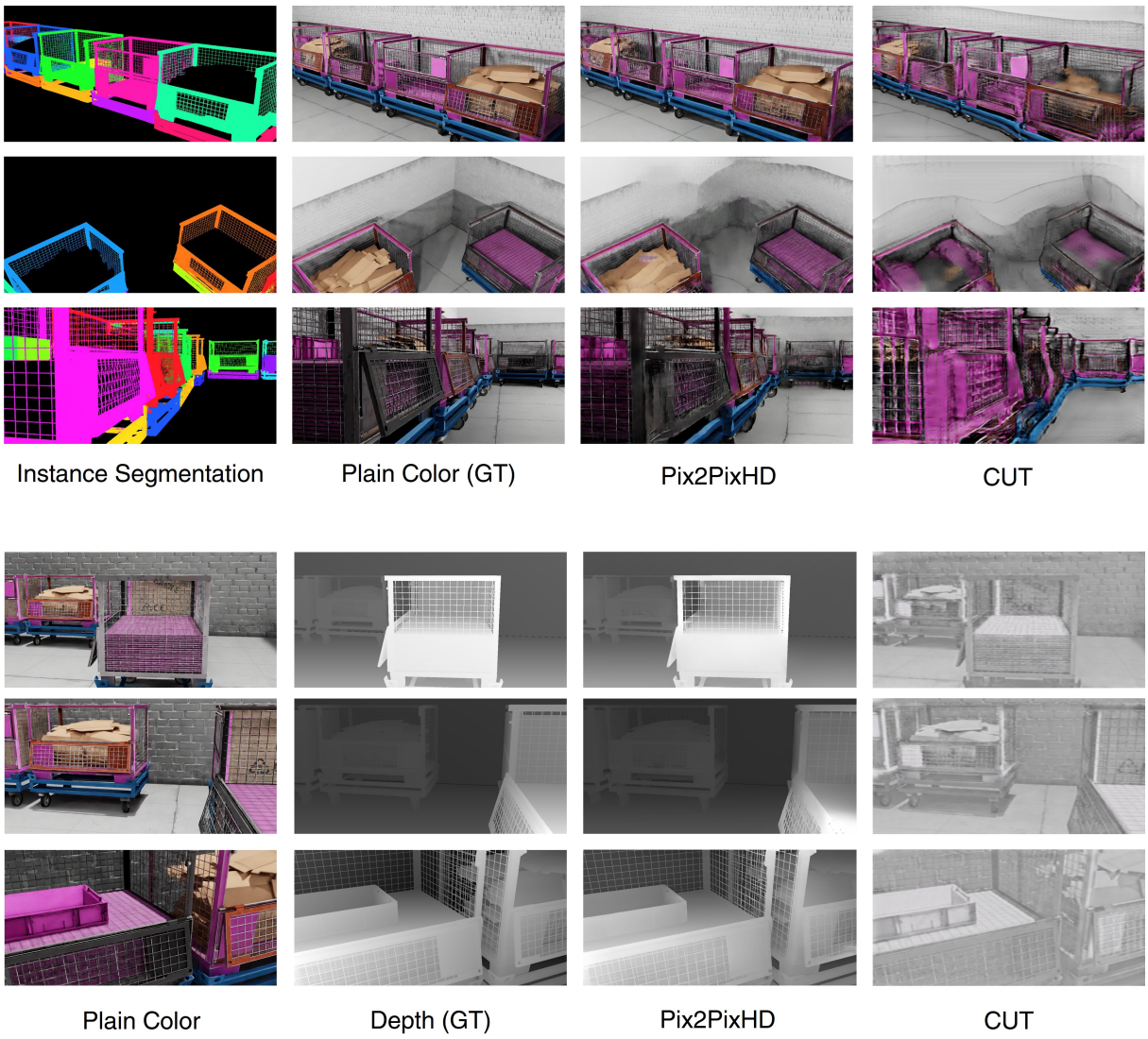


Fig. 19 Comparing domain transfer for colorization and depth estimation tasks using Pix2PixHD and CUT (200 epochs and 1500 paired images: 3000 total images)

color image from a depth image, highlighting the importance of single-modal output applications. The multi-modal output approach enables data augmentation by associating a single source image with a distribution of different outputs in the target domain, i.e., one-to-many associations (Chen and Jia, 2021) (cf. Figure 18), for example, coloring a single outline drawing with multiple combinations of colors, textures, light settings, and backgrounds. Unlike the mode collapse problem (Goodfellow, 2016; Durall et al, 2020; Mao et al, 2017), where multiple inputs are mapped to the same generated image, some multi-domain translation architectures, such as BicycleGAN (Zhu et al, 2017b), and PixelNN (Bansal et al, 2017), address this issue by using GAN mode collapse solutions and nearest neighbors approach respectively, to generate multiple outputs for a single input sample. Additionally, authors in (Denton et al, 2017; Kim and Mnih, 2018; Chen et al, 2016; Gonzalez-Garcia et al, 2018) adopted disentangled representations to generate different style variations while preserving the content, thereby avoiding unrealistic combinations.

Experimentation: In figures 19, I10 and I9, we

demonstrate the translation of images from the plain color domain to the depth domain and from instance segmentation images back to plain color, respectively. The models were trained using a dataset from a single scenario in both a supervised manner (using Pix2PixHD) and an unsupervised manner (using CUT as described in Section 6.2). Results show that supervised models outperform unsupervised ones compared to ground truth images. In the Pix2PixHD depth translations, small blob artifacts and dark lines were observed in high-detail areas, but the visual quality remains acceptable. For image colorization in Figure I9, Pix2PixHD maintained the original colors of the stillages while CUT, in most cases, adopted the same pink color for all assets. Furthermore, fewer details and more wavy lines were noticed in the unsupervised translations.

Limitations: However, despite the advantage of generating/translating complex scenes/modalities, a supervised two-domain I2I translation training necessitates a large number of paired data: Each image in domain A (source domain) must be associated with its corresponding image in domain B (target domain). This makes acquiring a supervised paired training dataset challenging and prone to human errors, as it requires a deep understanding and association of images from different domains (Chen and Jia, 2021). In contrast, unsupervised I2I only requires a larger dataset for each domain (Mustafa and Mantiuk, 2020), making it a broader and more robust branch of scientific research, particularly when creating mapping relationships between different domains while preserving high-quality and realistic translated images.

6.2 Unsupervised Training

In recent years, various methods have been proposed to address the lack of supervised paired data in image-to-image translation tasks. These include DualGAN (Yi et al, 2017), DiscoGAN (Kim et al, 2017), CycleGAN (Zhu et al, 2017a), UNIT (Liu et al, 2017), SCAN (Li et al, 2018), and U-GAT-IT (Kim et al, 2019b), among others. One common approach used in these methods is the **cycle-consistency constraint**, which involves translating an image x from a source domain A to a target domain B ($x_A \rightarrow x_B$) and vice versa ($x_B \rightarrow x_A$), and then comparing the original source image x_A to its reconstruction from the target domain x_{ABA} (i.e., $x_A \rightarrow x_B \rightarrow x_A$) (Zhu et al, 2017a; Chu et al, 2017).

Limitations: However, this constraint is inefficient when both domains are heterogeneous[13]. This is because the cyclic loss forces the model to generate in the translated image all the information present in the source image, e.g., keeping the beard in male-to-female face translations, which can lead to the preservation of irrelevant textures or the inability to remove or change large objects or shape (Zhao et al, 2020c).

As a solution, researchers in the field of I2I have proposed various architectures to overcome the limitations of the cycle-consistency constraint, which can be inefficient when dealing with heterogeneous domains. These architectures focus on maximizing the common features, information, and semantics between the source and target images: One approach is to modify the discriminator based on semantic segmentations, as proposed in GANimorph (Gokaslan et al, 2018). Another approach is to use a Siamese network to compare image similarities between both domain images, as proposed in TraVeLGAN (Amodio and Krishnaswamy, 2019). TransGaGa (Wu et al, 2019) disentangles domains into a Cartesian product, while ACL-GAN (Zhao et al, 2020c) replaces the cyclic loss function with an adversarial-consistency loss. Other architectures have focused on translations **beyond the cycle-consistency constraint**. These architectures take advantage of (i) insusceptible semantic information towards geometric transformations (Fu et al, 2019), (ii) equal distance between two source images and their translation images (Benaim and Wolf, 2017), (iii) self-similarity to represent a scene structure (Zheng et al, 2021), or (iv) contrastive learning to ensure a one-side translation process (Park et al, 2020a). Examples of these architectures include GcGAN, DistanceGAN, F-LSeSim, and CUT, respectively.

Unlike CycleGAN, CUT (Park et al, 2020a) does not use hand-crafted loss or inverse network. In addition to adversarial learning, CUT implements a contrastive learning-based framework to

[13] Cyclic loss best practices are manifested in small domain gaps: horses to zebras, summer to winter, etc.

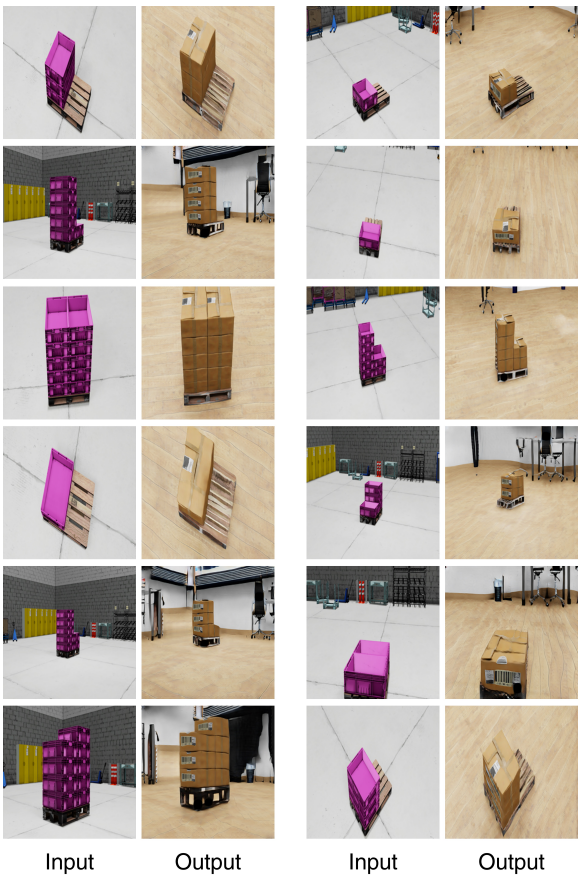


Fig. 20 Domain transfer from industrial (Domain A) to office (Domain B) environment assets (245 epochs and approx. 10,000 unpaired images: 20,000 total images)

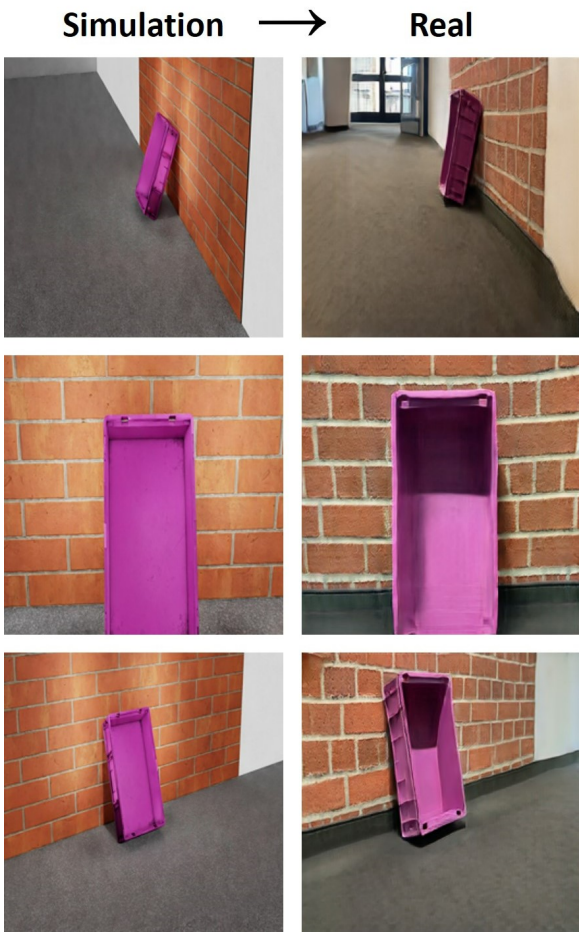


Fig. 21 Simulation-to-Real translation using CUT

maximize the mutual information between input (source) and output (target) domains (taesungp, 2020). *"Contrastive learning is a popular form of self-supervised learning that encourages augmentations (views) of the same input to have more similar representations compared to augmentations of different inputs"* (Saunshi et al, 2022). Taken two images from both domains, i.e., input and output, CUT considers the following: a pair of (1) positive pair (z, z^+), and (2) negative pairs (z, z_i^-), where z refers to an output patch, z^+ a similar patch to z in the input image and sharing common features at a same similar location: edge, shape, pattern, etc., and z_i^- are different patches from other locations of the input image. Afterward, CUT maximizes the cosine similarity of the positive pair while minimizing it between the negative pairs. This approach allows for faster model training and lower memory consumption, and has been shown to produce impressive results, such as translating images of pink KLT boxes in an industrial setting into cardboard boxes in an office environment (Figure 20), or to reduce the simulation-to-real gap and transferring rendered images into a realistic domain (Figure 21).

Limitations: However, it should be noted that CUT is only a one-sided I2I model. Furthermore, authors in (Junyanz, 2017) note that both domain datasets must share common visual content. Otherwise, it fails on random combinations.
Additionally, since the contrastive loss uses internal patches of the same image, the CUT model can be extended to single-image training, where a single image represents each domain. Additional details are provided in the next section.

Yet, SOTA approaches for I2I translation have focused on transferring the whole style

of the source image without considering fine or local object translations. Several approaches have been proposed for **fine-grained object translations** to address this limitation. These include approaches which (i) only translate segmented objects while keeping other regions, such as backgrounds, intact (Mo et al, 2018), (ii) attention GANs and attention-guided I2I methods which focus on individual objects (Chen et al, 2018; Alami Mejjati et al, 2018), and (iii) instance-aware I2I approaches for fine-grained local instance manipulation (Shen et al, 2019). Additionally, many researchers have extended the CycleGAN from one-to-one mapping to one-to-many or many-to-many mappings for multi-modal outputs (Kazemi et al, 2018; Almahairi et al, 2018). Furthermore, disentangled representations have been proposed as a solution for multiple generations (Lin et al, 2018; Huang et al, 2018; Lee et al, 2018; Ma et al, 2018a). Finally, other studies have used mode collapse solutions to generate more diverse image translations (Mao et al, 2019).
Limitations: However, Chang *et al.* have pointed out in (Chang et al, 2020) that disentangling the representation of a domain-invariant content space breaks the relationship between the image content and style.

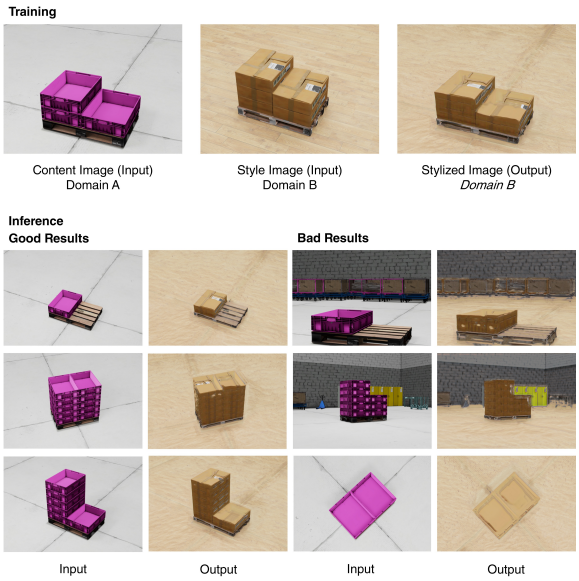


Fig. 22 SinCUT: Transfering KLT Box into Cardboard Box (90 epochs, 35 hrs approx.)

6.3 Semi-Supervised and Few-Shot Trainings

Collecting a large training dataset in industries is a difficult task. However, the literature includes several approaches for training I2I translation models with limited labeled data or even a few training images. For example, Mustafa *et al.* introduced the concept of transformation consistency regularization (TCR) in (Mustafa and Mantiuk, 2020) and used it to train image colorization, image denoising, and image super-resolution models with only 10% labeled data as a form of **semi-supervised training**. Additionally, transfer learning and domain adaptation techniques, such as Transferring GAN (Wang et al, 2018c) and MT-GAN (Lin et al, 2019), and EWC (Li et al, 2020d), can also be used to train I2I translation models with limited data, by leveraging pre-trained models that were trained on a large source domain to target domain dataset.
Another approach is one-shot I2I, which is an extreme form of few-shot I2I, where a model is trained using only one source image and a set of target domain images, such as OST (Benaim and Wolf, 2018) and BiOST (Cohen and Wolf, 2019) that are based on weight sharing strategy with selective backpropagation, and feature-cycle consistency respectively or (ii) two unpaired images for two domains, e.g., TuiGAN (Lin et al, 2020) that is based on progressive translation technics or SinCUT (Park et al, 2020a), an extension of the pre-mentioned CUT architecture as shown in Figure 22. These approaches are useful for training models using hardware on a budget or simple GPUs. Moreover, to achieve good results with SinCUT, the author recommends the following on his official CUT GitHub repository (taesungp, 2020):

1. *"It's very important that the target reference image has a similar structure as the source. Choosing a suitable target reference image might be a nontrivial problem."* Because, SinCUT is not able to extract general knowledge of a domain that is represented by a single image. - Issue #9
2. Patience is needed. The training requires some hours. - Issue #51
3. Adopting lower resolution images is a solution to avoid *CUDA out of memory* problems. - Issue #125

Table 3 GAN Image Translation Approaches' Brief

I2I Approach	Advantages	Drawbacks	Datasets
Supervised	Translates complex scenes. Supports multimodality outputs by adopting collapse mode solutions and features disentanglement	Requires hard data acquisition and pre-processing to pair data from different domains	Cityscapes, CMP Facades, *Google Maps Scrapped Images*[a], Zappos shoes, edge2shoes, edge2handbags, NYU Indoor RGBD, ADE20K, Helen Face, night2day, Oxford-IIIT Pet dataset, Deepfashion (HD), Dayton, CVUSA, Surround Vehicle Awareness (SVA), Ego2Top, Radboud Faces, NTU Hand Digit, Senz3D, Market-1501
Unsupervised with Cyclic Loss	Translates images without any supervised paired data. A bidirectional translation can be supported.	Best performs with small and simple changes between both domains. Requires a larger dataset	Photo-sketch, Day2Night, Facades, *Aerial-maps*[b], Fidler's 3D Cars[c], Facescrub, CelebA[d], Cityscapes, Maps2Aerial, Edges2Shoes, Horse2Zebra, Apple2Orange, Summer2Winter in Yosemite, Painting2Photos (Monet, Cezanne, Van Gogh, Ukiyo-e), SYNTHIA[e], ImageNet[f], iPhone2DSLR Flower, Selfie2Anime
Unsupervised without Cyclic Loss	Translates images between 2 domains with significant gaps, e.g., shapes	Translates into the global style target domain without considering fine-grained features and local information	CelebA, Selfie2Anime, **Cat2Human**, **Human2Dog**, **Giraffe2Horse**, **Cheetah2Cow**, **Lion2Rhino**, **Bear2Wolf**, Horse2Zebra, SVHN, MNIST, **Car2HumanHead**, Edges2Shoes, Blond2Black hair, Male2Female[g], **Handbags2Shoes**
Unsupervised Fine-grained	Translates local fine-grained object without impacting global coarse objects	Additional information e.g. segmentations might be needed to ensure a fine-grained object translation	CCP, MS COCO, MHP, Photograph2Portrait, Cat2Dog[h], Facades, Summer2Winter, CUB, Streetscape, Lion2Tiger
Semi-supervised & few-shot	Requires fewer data, labels, and hardware	Less accurate than previous training. For better performance, single-image domains must satisfy special conditions.	Places, BSD, LSUN, ImageNet, CelebA, Flowers, LFW, Artistic-Faces, MNIST, SVHN, Paintings2Photo, AFHQ

[a]Zhou *et al.* argues in https://bland.website/city2city/final_report.pdf that using images specifically and uniquely related to a defined area, such as Paris StreetView, is superior to using randomly selected images, like those found in Google Street View Images, for a City2City translation.

[b]self-captured from Google Maps: https://github.com/duxingren14/DualGAN

[c]DiscoGAN used rendered images of 3D Cars (https://www.cs.utoronto.ca/~fidler/projects/CAD.html) and 3D Faces dataset from Paysan *et al.* (Paysan et al, 2009): https://github.com/SKTBrain/DiscoGAN

[d]It is used for gender translation, or to add attributes such as blond hair, smiling, eyeglasses, etc.

[e]Used with Cityscape for Sim2Real translation

[f]Translation between different dog breeds, cat breeds, etc.

[g]a lower domain gap indeed exists compared to other experimentation datasets, but the problem remains in preserving the source image identity after style translation (Chang et al, 2020).

[h]Such datasets can be used in both translation directions, e.g., Dog2Cat as well (Chang et al, 2020)

As shown in Figure 22, we trained a model for 90 epochs, with 100,000 iterations per each epoch. The training process took 35 hours on a 24 GB NVIDIA GeForce RTX 3090 GPU. Upon testing the model, we observed that the quality of the images improved when they shared the same camera angle as the training image. However, when the camera angle was different, the KLT box shape was still recognizable but had a different texture.

6.4 Multi-domain I2I Translations

From another perspective, many researchers focus on multi-domain I2I translations, which involve

a single unified network that handles many-to-many relationships between different domains, such as a single complex dataset with multiple subclasses. This approach eliminates the need to train $n \times (n-1)$ two-domain I2I models to achieve the same goal, reducing maintenance complexity and saving time and memory. Similarly to previous taxonomies, multi-domain I2I architectures offer unsupervised, semi-supervised, and few-shot training methods, as highlighted in (Chen and Jia, 2021). One example of this is StarGAN, which builds on top of CycleGAN and is considered one of the most classic methods for multi-domain I2I translations (Choi et al, 2018). However, other architectures share the same mindset but are beyond the scope of this review. For more information, we recommend reviewing (Pang et al, 2021).

6.5 Summary

In Table 3, we review GAN-based I2I translation learning taxonomies, including supervised, unsupervised, semi-supervised, and few-shot approaches. In general, it can be concluded that each taxonomy excels in specific conditions and constraints.

7 Other I2I Applications

In this section, we project notable I2I applications in the industrial area, such as image de-filtering, image expansion, and super-resolution.

7.1 Image De-Filtering and Artifact Reduction

I2I translation includes image de-filtering applications such as deblurring images. We consider multiple blurring filters. For instance, motion blur is one essential and famous image blur effect. It is manifested when the camera[14] or the target asset is moving. Kupyn *et al.* introduced DeblurGAN as a solution to restore images. The conditional WGAN inspires the architecture with a gradient penalty and a perceptual loss. Such architecture is heavily adopted in other I2I translations as colorization, inpainting, dehazing, etc. However, the training necessitates a supervised paired dataset where the source and target domain contains blurred and sharp images. For our experimentations, we imitated the output of some camera-defected shots such as motion blur for a moving camera, desaturation as a camera with a defected or old sensor, foggy and hazy environments with water steam or smoke (Zhang et al, 2017b), and noises and blobs as a dirty lens. Processed and restored images are displayed in Figures 23 and I11.

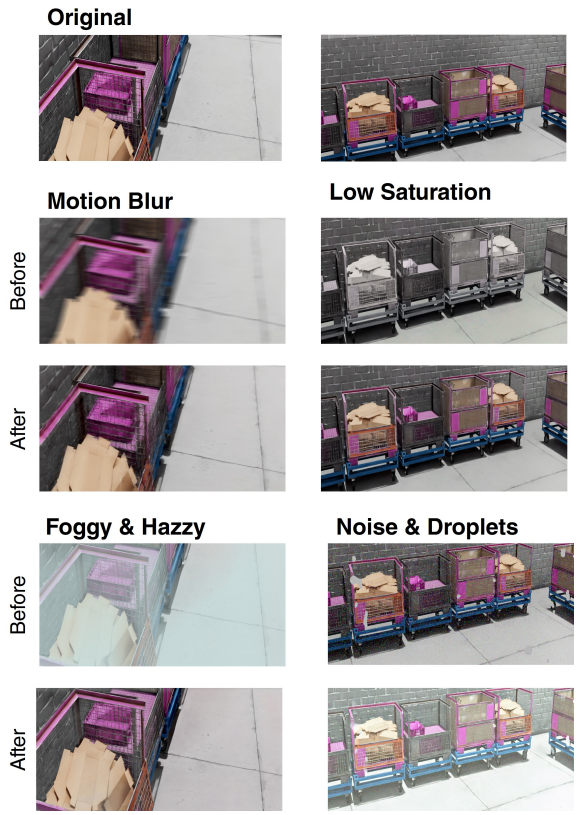


Fig. 23 Image deblurring, saturating, dehazing, and denoising

This architecture performs fascinatingly as long as the training dataset is adaptive and specialized to a single type of filter, e.g., motion blur exclusively or Gaussian blur. Consequently, integrating de-filtering models in the inference phase before robot actions can increase the prediction precision and optimizes a robot's decision-making behavior. Additionally, GAN I2I applications were proposed in hybrid with traditional reduction methods to efficiently reduce visual artifacts such as metal artifacts in medical captures (Gomi et al, 2021),

[14] E.g. the camera can be set on a transport robot or a moving robot arm

or saturation artifacts caused by reflective surfaces (Liu et al, 2021).

7.2 Image Expansion

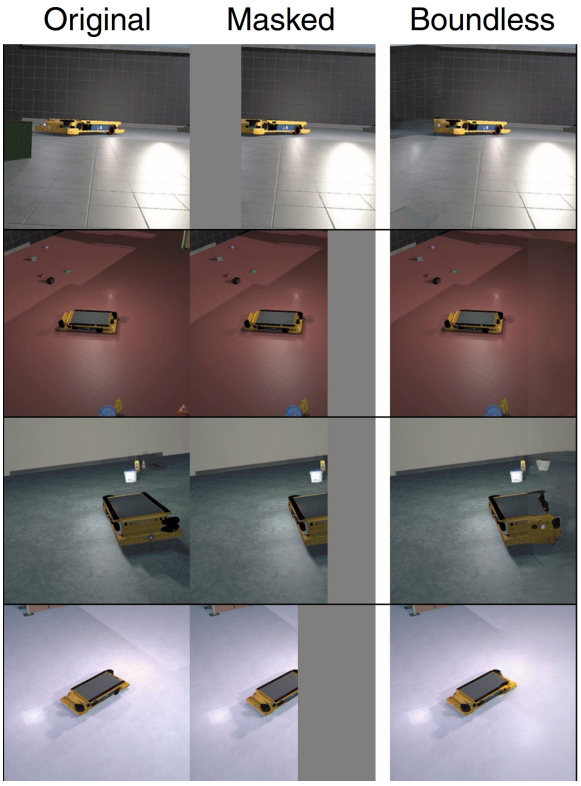


Fig. 24 Image expansion using Boundless

Some of the aforementioned GAN architectures downscale the training images multiple times by a factor of two, e.g., 128×128, 256×256, 512×512, etc. For this reason, (1) the training dataset is usually cropped, resized, contracted, or stretched, and this leads to data loss or distortion. This impacts the generated data, necessitating a post-processing phase for restoring and adjusting data. (2) Other suggestions depend on padding images with equidistant black filling on the top-bottom and left-right of the image to reach an image shape divisible by two multiple times. Nevertheless, the GAN learns about the border, and the generated square images are also subject to post-processing.
Image inpainting is a solution for processing training and generated images. Any training or generated image is subject to extrapolation (1) to satisfy the square shape training dataset processing instead of distorting the initial image or adding random color padding (2) to provide a generated dataset with a rectangular shape, respectively. However, applying image inpainting can be challenging, especially since existing SOTA techniques inpaints (1) blurry or (2) repetitive pixels. As a solution, GAN-based image inpainting surpasses the inconsistent semantic filling and shows impressive and promising results as in (Wang et al, 2019). Similarly, Teterwak *et al.* propose in (Teterwak et al, 2019) boundless GAN for image extension but with semantic conditioning to the discriminator. Internally, Boundless takes a square image; it masks some portions and tries to complete the masked part. E.g., to extend an image of $l \times w$ dimension with $x\%$ in width, it is possible to (1) crop $l - (x \times w)$ from the desired edge, (2) collage it with a $x \times w$ rectangle mask of the same length, (3) infer using the pre-trained model and finally (4) add the expanded part to the original image. As output, we found that BoundlessGAN in Figures 24 and I12 did not only expand images by expanding a ground or a wall texture, but also it created new assets in the background and learned asset shapes and compositions, e.g., no matter of the perspective and camera angle, it recovered a cropped STR with the proper dimension and generated its black LIDAR sensor in the correct position as well.

7.3 Super Resolution

Most GANs generate square images with a scale to the base of two but are limited to a maximum of 1024 px. For complex scene images with hundreds of assets, fine details assets, or small-size components, GAN cannot present such details. Therefore, cropping input images into sub-images is a possible solution to reduce scene complexity. In this case, super-resolution (SR) increases (1) the original dataset resolution before preprocessing, e.g., cropping, sampling, etc., or before training for better-generated image quality, or (2) the output quality immediately for faster training and generation, and sharper details.
In SR, the GAN's discriminator distinguishes the generator's SR images as real high-resolution images or artificial ones (Li et al, 2021). SRGAN is the first GAN implementation for SR. It supports scaling images by four (Li et al, 2020a).

SRGAN applies multiple loss functions: (1) MSE loss for pixel similarity, (2) perceptual loss based on VGG, a deep CNN network to capture image features at different scales to obtain details, and (3) standard GAN adversarial loss (Ledig et al, 2017). In addition, The "peak signal-to-noise ratio (PSNR)" is a pixel-based metric that expresses the ratio between the maximum signal value and the noise distortion power. The PSNR loss function is widely used for construction quality evaluation. Therefore, the higher the PSNR is, the better the image quality is (Johnson, 2006; Li et al, 2021):

$$PSNR = 10 \times \log_{10} \left(\frac{L^2}{\frac{1}{N} \sum_{i=1}^{N} \left(I\left(i\right) - \hat{I}\left(i\right) \right)^2} \right) \tag{1}$$

where L is the maximum pixel value, N is the number of pixels, I and $\hat{I}$ are the GT and reconstructed image respectively.

However, when applied in SR GAN-based methods, SOTA experimentation presents a significant opposite interpretation for the PSNR value. Contrary to the SR CNN-based methods, researchers noticed that the PSNR and the perceptual image quality are inversely proportional: a higher PSNR results in a smoother image; therefore, a lack of realism in details (Li et al, 2020a, 2021; Anwar et al, 2020). It does not consider any structural information in the images. This emphasizes the use of perceptual loss as an alternative.
Researchers successfully continue optimizing SRGAN in all following released GAN-based SR methods. For instance, (1) SRFeat (Park et al, 2018) proposes two discriminative networks for image and feature domains to generate real texture (high frequency) information instead of noise artifacts. Moreover, the authors tested their framework to upscale by four times 74×74 ImageNet dataset (Anwar et al, 2020). Yet, for optimal performance, feature GAN and perceptual similarity losses' layer should change depending on the image content (Park et al, 2018). (2) Cycle-in-Cycle, i.e., CinCGAN (Yuan et al, 2018), is based on the cyclic loss and implements 2 CycleGANs: the first network transfers a low resolution (LR) image into a clean: no blur and no noise, space domain. Afterward, the second network consists of a pre-trained SR model, and it upsamples the clean LR to a high-resolution (HR) image (Wang et al, 2020c), (3) ESRGAN (Wang et al, 2018b) improves SRGAN's network by enforcing residual learning and adding dense blocks between the input and the output. Moreover, ESRGAN improves adversarial and perceived loss functions and the discriminator network to remove unpleasant artifacts and learn better texture and sharper edges, respectively, and (4) Real-ESRGAN (Wang et al, 2021b) replaces ESRGAN's VGG-style discriminator with U-NET structure to stabilize the training. It also presents a "higher-order" degradation model to overcome common ringing and overshoot artifacts and restore better texture details: LR data generation consists of 2 degradation orders, including blurring, downsampling, resizing, noising, and JPEG compression. A 2D sinc filter follows the second order to synthesize common ringing and overshoot artifacts, as shown in Figure 27. Real-ESRGAN is trained with only synthetic data and restores most real-world images with better visuals. Additionally, Wang *et al.* marked Real-ESRGAN+[15] model, which is trained on sharp GT images. The authors found that sharpening ground-truth training images balances a better sharpness and a lower overshoot artifact. Great restoration details are presented in Figure 25 below. We noticed the high fidelity on several levels, as the texture, geometric shapes, edges, the stillage's delicate wires, etc.

Yet, limitations persist when restoring some text, barcodes, labels, or human face images. In Figure 26, we tried to upscale text and barcodes at different sizes. However, it was not as efficient for the shapes and textures. Additionally, Wang *et al.* extended current ESRGAN architecture to support up to $\times 4$ scale, including. $\times 1$ and $\times 2$. Yet more research is needed to support higher scales or specify the optimal scale ratio.
However, the inference model is functional for higher scales. But, in this case, we noticed that iteratively inferring a model achieves sharper, cleaner, and better quality HR images than inferring it at once. For instance, RealESRGAN_x4plus (xinntao, 2021) is trained on $\times 4$ downscaled dataset. Therefore, for an upscale of 16, inferring

[15]The '+' sign normally denotes that model results are improved (Li et al, 2021).

Fig. 25 Real-ESRGAN+ ×4 upscaling **(Zoom in for best view)**

the model twice with its default settings is more optimal than inferring it once with a 16 upscale (check Figure 28). Although, it is possible to face a CUDA memory allocation problem while iteratively upscaling an image because the input image is successively turning bigger in byte size. As a solution, Real-ESRGAN suggests cropping the input image into several tiles so that each tile is processed separately. Afterward, all tiles are stitched together.
We compared different GAN-base SR technics in Table H5 for additional information.

8 GAN Failure Modes & Evaluation

In this section, we list some of GANs most essential failure modes, and evaluation metrics.

8.1 Failure Modes

Training GAN is problematic because, basically, it is about training both the generator and the discriminator in a zero-sum game. This means that the improvement to one model could come at the expense of the other model. However, monitoring the training loss functions is insufficient to assess a GAN performance. More details are in the next Section 8.2. Thus, GAN failures do not result only during the training phase as mode collapse, training divergence, or class leakage. But in the generated image quality, even if the training was "quantitatively" successfully stable, e.g., image artifacts, they are mainly caused by bad signal processing, in other terms, the GAN architecture itself. Although GAN relies on two neural networks (NN), traditional problems may still occur as the vanishing gradient or training overfitting.

8.1.1 GAN-related failures

- **Mode collapse**: *i.e. catastrophic collapse, or the Helvetica scenario.* In fact, the generator always seeks a single output that looks the most plausible for the discriminator. GAN's generator may be stuck in a local minimum, which always produces the same plausible output image for any latent vector input. In its turn, if the discriminator also gets stuck in a local minimum, it will not reject the same generated image (Google Developers, 2022). At this moment, the GAN is in total or partial[16] mode collapse, and it omits all or portions of the target instances and distributions, respectively (Bau et al, 2019). The mode collapse can occur in two forms: intra-class or inter-class mode collapse, where GAN produces the same image for a single class or all the classes

[16] GAN may succeed in generating some classes while it fails in covering all samples for other classes.

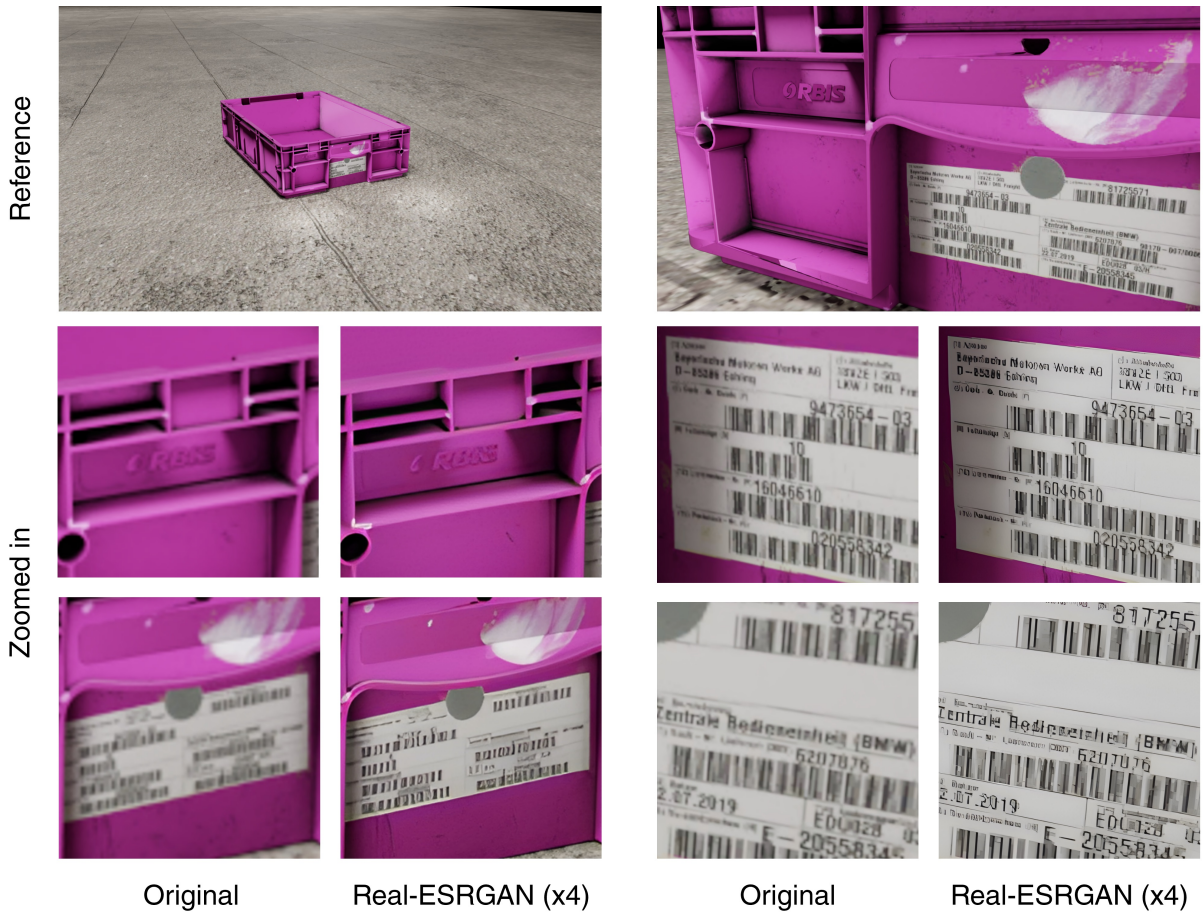


Fig. 26 Real-ESRGAN+ ×4 upscaling for text and barcodes **(Zoom in for best view)**

respectively (Saad et al, 2022) (check Figure 11). In fact, conditional GAN is more vulnerable to mode collapse (especially intra-class mode collapse) (Boulahbal et al, 2021) and degrades faster than unconditional training because the training dataset diversity is often divided over all classes, resulting in a lack of intra-class variation with limited changes in a limited size dataset. Taken together, it may lead to mode collapse (Shahbazi et al, 2022). The possible attempts to remedy consist of adopting different loss functions for both GAN networks (Google Developers, 2022), or reducing the learning rate. Moreover, Shahbazi *et al.* suggested in (Shahbazi et al, 2022) a new training strategy for cGANs by starting with unconditional GAN training and gradually leveraging the training by injecting class conditioning into the generator and the objective function. On another side, the training dataset is the leading player in GAN training. Therefore, we present more details concerning dataset manipulation in Section 4.1.2.

- **Convergence failure / Training divergence**: A GAN does not reach an utopian stable convergence state. Instead, it is often fleeting, as shown in figure 29. Indeed, when the generator perfectly succeeds, the discriminator has a 50% accuracy, i.e., it "flips a coin" to make its prediction. After this point, the discriminator feedback becomes less meaningful over time. If we continue our GAN training, the discriminator will end by returning random junk feedback affecting the generator performance negatively, and its quality may collapse (check Figure 13) (Google Developers, 2022).
- **Replicas generation**: It is a pattern of a repetitive set of pixels. It is prominent in

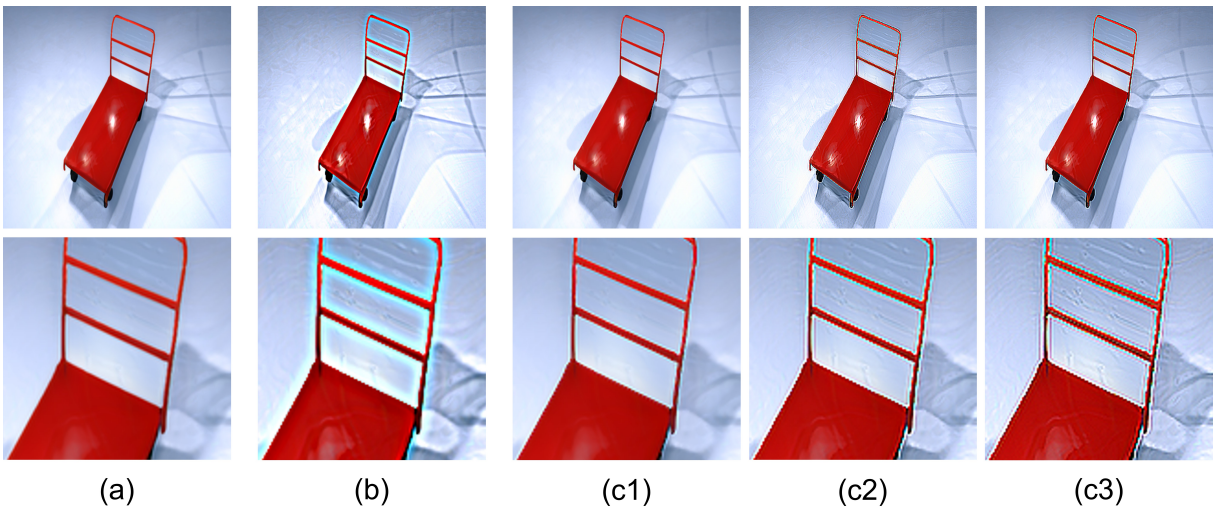


Fig. 27 (a) Initial image (b) Unsharp Mask (sinc): Overshoot is the first step of the ring where it is the most accentuated. Afterward the signal overcorrects itself and is below the target signal. The phenomenon is oscillative, leading to ringing artifacts (fainted ring) (c1) Sharp filter causing the first overshoot ring (c2) Second sharp filter causing a second overshoot ring (c3) Third sharp filter leading to a third overshoot ring **Zoom in for best view**

single-image training or while expanding images (check GramGAN in Section 5).

- **Mode connecting:** It refers to the difficulty of the generator in producing a wide range of output images that cover the entire target distribution. Especially when the data distribution is supported on a set of disconnected data manifolds in a very high dimension space. Therefore, the generator - a continuous function - may either discard some of the data manifolds (a form of mode collapse) or try to connect the manifolds (the mode connecting problem) (Armandpour et al, 2021).
- **Class leakage**: It is when the generated image inherits from different class features and styles, resulting in a newly created mixed object as shown in Figure 8. This occurs when the model is partially trained (Brock et al, 2018). Furthermore, it was noticed in some unconditional generated images where the discriminator suffers from a lack of conditionality (Boulahbal et al, 2021) (check Section 4.1.2). This failure is mainly perceived at the inference level.

8.1.2 NN-related failures

They can usually be detected while monitoring the networks' loss functions:

- **Vanishing gradient**: Authors in (Arjovsky and Bottou, 2017) suggest that if the discriminator is too confident, the generator may fail due to vanishing gradients. Vanishing gradient *"refers to the fact that in a feedforward network (FFN), the backpropagated error signal typically decreases (or increases) exponentially as a function of the distance from the final layer"* (Sussillo and Abbott, 2014). Hence, proper information cannot propagate from the output end to the layers near the input end. Therefore, this problem limits the development of GAN's generator network, resulting in GAN instability. Possible solutions rely on considering a Wasserstein loss function or a modified minimax loss. Although, an optimal discriminator does not provide all information for the generator for its progress.
- **Exploding gradient:** Opposite to the vanishing gradient problem, and may cause GAN instability and therefore to a mode collapse (Tao and Wang, 2020).
- **Training overfitting**: As previously mentioned, the learning process of GAN models alternately trains the generator and discriminator successively. However, when the discriminator excessively depends on the training data, the generator generates synthetic images that appear similar to the learning images. This is what we call "GAN overfitting" problems. At

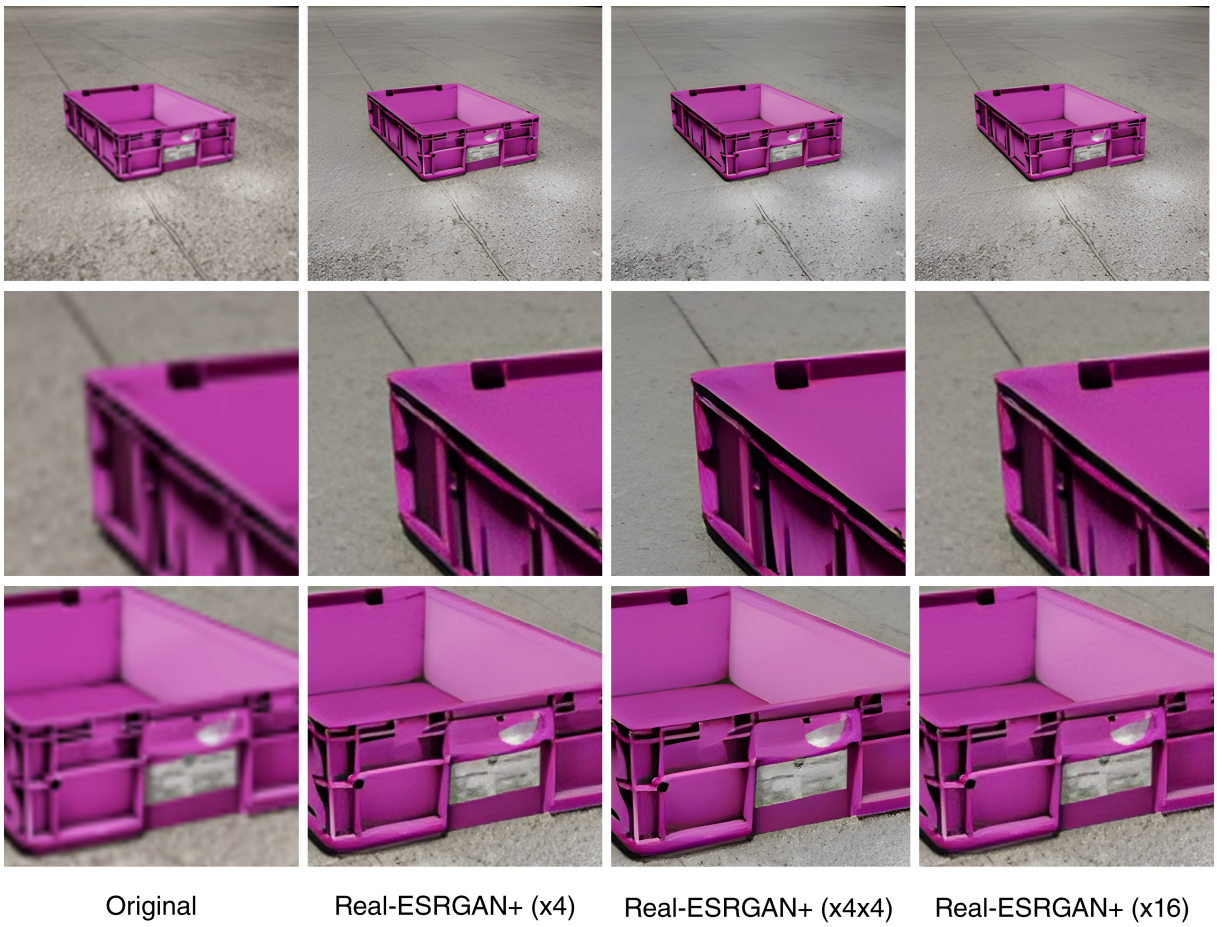


Fig. 28 Iterative Real-ESRGAN+ ×16 upscaling **(Zoom in for best view)**

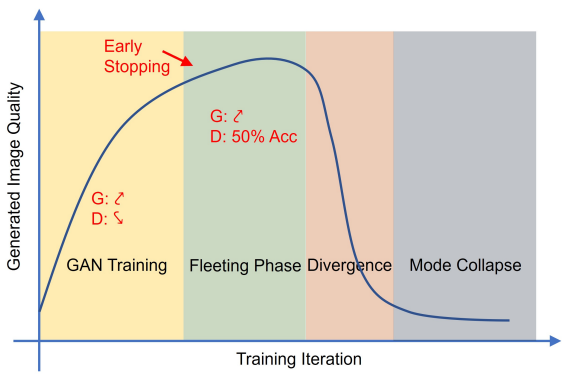


Fig. 29 GAN life cycle

this point, the GAN loses its meaning of data augmentation (Kim and Park, 2022).

8.1.3 Image Artifact

They are all perceived at the inference level. Authors proposed an Artifact Index algorithm to detect specific "physically-defined" artifacts (Wang et al, 2013; Gomi et al, 2021). From another side, other authors suggested detecting these artifacts from the frequency spectrum of the generated images (Dong et al, 2022; Zhang et al, 2019b).

However, we can return their causes either to the adversarial training where the generator adds some additional noise or features to fill a gap and fool the discriminator, or the adopted generation method when synthesizing the image.

Adversarial training based artifacts:

- **Noise artifact**: It appears as high random noise in synthetically upscaled images. It is

because real high-resolution image contains high-frequency information and details, contrary to super-resolution generated images. Therefore, the generator fools the discriminator by adding random high-frequency noise (Park et al, 2018).

- **Saturation artifact**: It results in high saturated generated images. As explained in BigGAN (Brock et al, 2018), truncating the latent space by re-sampling values above a certain threshold to zero provides a boost for Fréchet Inception Distance (FID) and Inception Score (IS) evaluation scores. Still, it degrades large models' generation quality by increasing the saturation channel. High-saturated images could lead to inaccurate color-based detection and decision-making.

Generation-based artifacts:

- **Aliasing artifact**: One of the main visual symptoms of an aliasing artifact is the visual stair-stepping that occurs on the edges, as shown in Figure 10. It is caused because of careless signal processing while sampling. Karras *et al.* addressed the aliasing problem in StyleGAN3 (Karras et al, 2021).
- **Texture sticking artifact**: GAN does not hierarchically synthetize images. Thus, coarse features control finer detail features: Textures statically stick to the same coordinate and do not move with the corresponding objects. So, unwanted texture information (static layer) overlays the wrong object's details (dynamic layer) (Karras et al, 2021).
- **Water-droplet artifact**: It is a droplet-like shape with a blurry texture and occurs in different locations in the generated image. It was found that this noise originates from 64×64 feature maps and propagates into the output image (Karras et al, 2019). This issue was taken care of in StyleGAN2 (Karras et al, 2020b) by replacing the Adaptive Instance Normalization (AdaIN) with estimated statistics (check Section D.2).
- **Phase "mismapping" artifact**: It shows mismapping local fine-grain features to the global features. However, "motion" is the most common cause of placing data in the incorrect location during data collection. Karras *et al.* attribute phase artifact to the progressive nature of their StyleGAN (Karras et al, 2019, 2020b,a, 2021). High-frequency feature maps are collected and generated over each progression in the middle layers, leading to shift invariance.
- **Texture blobs artifact**: It consists of a shapeless, contiguous, or amorphous vague shape while preserving the right realistic texture or color. In other terms, it is a collection that lacks a definite shape. This was observed when the intended generation condition was not well covered in the training set (Brock et al, 2018) (check BigGAN in Section 4.1.1). Additionally, it could be an issue with the GAN architecture, e.g., DCGAN and modified-DCGAN, that cannot support a large dataset with a big number of object classes (Salimans et al, 2016).
- **"Local" checkerboard artifact**: It is due to the generation method rather than the adversarial training approach because it can be perceived at the very first random weight image sampling and in other generative methods than GANs. It is a pattern of alternating light and dark pixels of the same color, just like a checkerboard, from where it got its name. This artifact is prominent in images with strong colors and gradients. It is mainly caused by the deconvolution operation when the small size generated image is being upsampled: every point is scaled into a bigger square, creating, easily, uneven overlap where it puts more metaphorical paint in some pixels than others (Zhang et al, 2019b; Odena et al, 2016; Brock et al, 2018) (check Image 3). However, some checkboard artifacts may be issued from the gradients when deconvolution in the backward pass of the neural layers. Still, the gradient artifact topic is not very well studied in the SOTA (Odena et al, 2016).
- **Overshoot artifact**: It occurs in sharpened images as an increased jump around edges, where signals are bandlimited without high frequencies, e.g., after applying the sharping algorithm, e.g., unsharp masking, sinc filter, or JPEG compression, etc. (check Section 7.3).
- **Ringing artifact**: It appears as false edges near sharp transitions. They visually look similar to bands or "ghosts" near edges. Hence, the shape is outlined with multiple parallel bogus edges. Usually, it is combined with other types of artifacts, e.g., overshoot, clipping, etc., for

the same reasons (Wang et al, 2021b; Cao et al, 2011).

8.2 Evaluation Metrics

To the best of our knowledge, there is no fixed evaluation metric convention to evaluate the GAN model's performance. However, due to the rich research in GANs and the various applied applications, existing SOTA GANs mention multiple metrics that can be mainly classified under subjective and objective categories.

8.2.1 Subjective metrics

Subjective metrics are qualitative methods and are mainly based on human opinions. Despite the significant feedback from the manual inspection, reviewers must have good knowledge about the image ground-truth details and what does not belong to the target domain. This evaluating system is unscalable and cannot be implemented in an automated GAN training process. Thus, researchers tend to collect these reviews and train complex networks that try to imitate the reviewer's decision-making behavior, e.g., Inception v3, which is trained on the ImageNet dataset. Furthermore, Borji cites in his latest review (Borji, 2022) other approaches for subjective evaluations.

8.2.2 Objective metrics

Objective metrics are quantitative methods and are based on numerical comparisons such as:

1. **Mathematical formulas** e.g., Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) (Wang et al, 2004). They are pixel-level comparisons and do not consider any perceptual quality, image content, or semantics.
2. **Model-based evaluation**: In this case, pre-trained models imitates humans' behavior for opinion scoring, which is faster than any qualitative, subjective metrics. However, huge datasets are required to train such classification models. Many datasets exist, e.g., Berkeley Adobe Perceptual Patch Similarity (BAPPS) dataset consists of one reference image and two distorted images with human-based similarity judgments. Zhang *et al.* used this dataset in (Zhang et al, 2018c) to train Learned Perceptual Image Patch Similarity (LPIPS). In addition, the dataset can be collected from online surveys platform as Amazon Mechanical Turk (AMT) as well. Yet, the model highly depends on a subjective decision-making dataset. The two most widely adopted and essential metrics are Inception Score (IS) (Salimans et al, 2016), Fréchet Inception Distance (FID) (Heusel et al, 2017). Furthermore, multiple variants were published for IS, and FID enhancements (Borji, 2022). For instance, Single image FID (SIFID) compares two single images (Shaham et al, 2019), Spatial FID (sFID) considers spatial features (Nash et al, 2021), Fast FID speeds up the traditional FID (Mathiasen and Hvilshøj, 2020), Memorization-informed FID (MiFID) generates a more similar image to the training dataset (Bai et al, 2021), Unbiased FID/IS mitigates bias issues (Chong and Forsyth, 2020), Fréchet Video Distance (FVD) (Bhagwatkar et al, 2020), Fréchet Audio Distance (FAD) (Kilgour et al, 2019), and Fréchet ChemNet Distance (FCD) (Preuer et al, 2018) extends FID to evaluate generated videos, audios and molecules respectively. However, in backward thinking, Binkowski *et al.* introduced the Kernel Inception Distance (KID) checking, in a lower variance mode, the dissimilarity between both datasets' extracted features (Bińkowski et al, 2018).
3. **Data manifold-based analysis**: It interprets image similarity based on fundamental surfaces where the images are described and found. For instance, Barua *et al.* introduces Cross Local Intrinsic Dimensionality (CrossLID) to calculate the concurrence area between two data domain distribution's manifolds (Barua et al, 2019). Unlike inception distances, i.e., FID and KID, data manifold-based metrics such as the Intrinsic Multi-Scale Distance (IMD) (Tsitsulin et al, 2019) and Cross-Barcode (Barannikov et al, 2021), compare unaligned data manifolds and distinguish distributions at high-dimensional spaces and not only local but global structures as well, without relying on any pre-trained networks. For example, the cross-barcode metric is effective for multiple domains, e.g., images, time series, 3D shapes, and datasets. Faster and more sensitive than

Table 4 GAN Failure Modes

Failure	Category	Cause	Solution	Example
Mode Collapse	GAN-based	Generator and discriminator stuck in a local minimum	Changing loss function, e.g., Wasserstein loss - reduce learning rate - adopt a training strategy, e.g., starts with an unconditional GAN training and gradually introduces class conditioning - Ensure that the training dataset is diverse and has a sufficient variation - Implement stabilization technics and convergence improvement	Figure 11
Convergence Failure	GAN-based	Imbalanced Generator and discriminator, i.e., the discriminator is more powerful than the generator - High learning rate - Low variance training dataset	Right selection of training checkpoint - Implement stabilization technics and convergence improvement: history of generated samples, self-attention mechanism, spectral normalization method, etc. - Maintaining a balance between G and D: weight clipping, gradient penalty, etc.	Figure 13
Replicas Generation	GAN-based	Limited control to capture the full range of image styles and content variation from a small or single image dataset	-	Figure 17
Mode Connecting	GAN-based	High dimension of data distribution support on disconnected data manifolds	Switch to conditional GAN training (Armandpour et al, 2021)	-
Class Leakage	GAN-based	Mostly noticed in unconditional training where the generated samples are biased toward specific dominant classes	Ensure a good separation of the dataset in conditional training	Figure 8
Vanishing Gradient	NN-based	Discriminator is too good	Replace loss function with Wasserstein or minimax - Stabilize the GAN training, etc.	-
Exploding Gradient	NN-based	(Inverse of vanishing gradient) Large updates to the GAN weights leading to its destabilization with large error gradients	Stabilize the GAN training using zero-centered gradient penalty, two time-scale update rule, exponential moving averaging, etc. - Apply gradient alleviation (Tao and Wang, 2020)	-
Training overfitting	NN-based	Use of limited dataset - Training the GAN for too long	Limiting the discriminator learning by using Limited Discriminator GAN (Kim and Park, 2022) - Increase dataset variety	-
Noise	Adversarial	Added by G to fill high-frequency details	Adding more than one discriminative network to consider high-frequency data, e.g., textures (Park et al, 2018)	-
Saturation	Adversarial	Truncating and re-sampling values of the latent space to 0	Conditioning G to smoothly enforce amenability to truncate the latent space (Brock et al, 2018)	Figure 3
Aliasing	Generation	Careless sampling	Treating all signals as continuous with high-quality upsampling filters + architecture changes to be equivariant to sub-pixel translations and rotations (Karras et al, 2021)	Figure 10
Texture sticking	Generation	Border effects and aliasing problem	Alias-Free GAN (Karras et al, 2021) refines coarse input Fourier features into local oscillations dependent on the content	Figure 9
Water-droplet	Generation	Originates from 64×64 feature maps	Replace AdaIN with estimated statistics (Karras et al, 2020b)	-
Phase Mismapping	Generation	Caused by Progressive nature of StyleGAN	-	Figure 1
Texture blobs	Generation	Lack of data when using specific GAN architectures that support large datasets	-	Figure 2
Local Checkboard	Generation	Upsampling small size generate images after deconvolution.	Applying resize-convolution steps after replacing deconvolution steps with up-sampling (Odena et al, 2016; McCloskey and Albright, 2019)	Figure 3
Overshoot	Generation	Application of sharpening algorithm after super-resolution task	Sharpening ground-truth training images (Wang et al, 2021b)	Figure 27
Ringing	Generation	Combined with overshoot artifact	Related to overshoot artifact solutions	Figure 27

Geometry Score (GS) (Khrulkov and Oseledets, 2018), it detects mode-dropping, intra-mode collapse, mode invention, and image disturbance and transformation such as flipping, rotation, etc. (Barannikov et al, 2021).

4. **Precision and Recall P&R-based metrics**: While the precision measures the image quality and similarity to the real images, the recall highlights the generation ability to cover all the real images' instances (Sajjadi et al, 2018; Borji, 2022). P&R-based metrics show the trade-off between precision and recall to avoid bad quality and mode collapse, respectively. Recently, Naeem *et al.* and Alaa *et*

al. built on top of the P&R, and introduced Density and Coverage (D&C) (Naeem et al, 2020), and Alpha Precision and Beta Recall (α-P&β-R) (Alaa et al, 2022) metrics.

5. **Task-driven evaluation**: Researchers argue that if GAN can learn a dataset distribution, then the generated synthetic data should perform well in downstream tasks, i.e., training models using synthetic data and inferring back on real data. However, comparing synthetic-trained models to real data-trained models' performance for some pre-defined and well-known tasks is an efficient method to estimate the efficiency of the GAN model. For instance, some of the applications refer to classification (Ravuri and Vinyals, 2019a), distribution analysis (Xuan et al, 2019), and object or face recognition (Bashir et al, 2021).

Additionally, due to the various number of GAN applications and use cases, researchers have developed adaptive and application-specific evaluation metrics, e.g., perplexity (Jelinek et al, 1977; Borji, 2022) and caption score (Ding et al, 2021) for text-to-image applications, or Fully Convolutional Networks (FCN) score for specific I2I applications (Isola et al, 2017; Long et al, 2015). To summarize the above, the literature presents hundreds of evaluation metrics to assess image quality in hundreds of GAN applications. Depending on each application description, each metric outperforms in its way. Therefore, it is impossible to lead all our GAN reviews and compare their performance based on a standard single evaluation approach. Table 5 presents the strength and weaknesses of the different GAN evaluation metrics: The "lowest" and "highest" columns are related to the metric values for which the '+' sign indicates a better image assessment contrary to the '-' sign.

9 Prospects and Opinions

Based on this survey and our experimentation, we will mention some advice and essential points regarding the selection and usage of GANs in this section. Then, we will conclude with some questions about some research topics that we found important and would ease the whole training process if we had some existing answers or studies.

- The diversity and size of a dataset directly impact the performance of GANs. In cases where the dataset is small or less diverse, alternative taxonomies such as semi-supervised and few-shot learning techniques, transfer learning, or disentangled feature generation may be more effective. Although supervised learning produces better results and mitigates many GAN failures, acquiring paired or labeled data is more difficult. Unsupervised learning, on the other hand, requires a larger dataset, and data augmentation may not be effective and can lead to augmentation leakage.
- Texture synthesis is distinct from traditional image generation, with multiple types requiring different GAN architectures. Single-image GANs are particularly promising for texture generation and image translation with limited hardware resources, although they require significant training time. However, once the model is trained, evaluation is fast. On another side, iterative GAN evaluation, as seen in SR, produces better results than relying on a single-shot inference with high ratios. However, text and number generation remain challenging with most image generation and translation techniques, not just those based on adversarial methods.
- It is important to distinguish between capture and generation artifacts: the first is caused by hardware and preprocessing algorithms and are considered true features by the GAN, which will attempt to reproduce them. Generation artifacts, on the other hand, are produced by the synthesis network due to low data coverage or poor network configuration. And currently, there is no unified evaluation standard for assessing GAN generation quality or shape-based artifact indices. Therefore, perceptual evaluation is still necessary. Relying solely on numerical values to determine when to stop training a GAN model is ineffective, but providing the GAN with ample training time is recommended to avoid premature stopping. However, excessive training can lead to a degradation in image quality. As such, the stability period of the GAN should be manually identified to select the appropriate checkpoint.
- Fewer studies cover a complete study on how a training dataset distribution would affect a conditional GAN: should all class datasets be equal and balanced? Does one class dataset diversity affect another class generation? While training,

Table 5 GAN Evaluation Metrics

Method	Category	Lowest	Highest	Strength	Weakness
Opinion Scoring	Qualitative	-	+	Considered a reference for image similarity comparison. Suitable for all evaluations, including complex and sophisticated images	Time-consuming. Subjectivity and prone to human error. Professionals are needed to evaluate specific field data. Non-scaling system. It cannot be used for monitoring training steps.
PSNR	Quantitative	-	+	Mostly used. Based on MSE. Shows the intensity difference between translated and GT images.	Pixel-based and does not consider any structural information. Misleading in some applications: two visually different images can have a high PSNR.
SSIM	Quantitative	-	+	Mostly used. Supports change of luminance, contrast, and structural similarity comparison.	Unstable in low-variance area images.
IS	Quantitative, Model-based	-	+	Pretrained classifier based on Inception v3 trained on ImageNet. No human subjectivity consideration.	Most accurate for classes that the model was trained on. Supports small 300×300 px images. Reliable at large sample size. Evaluates only the generated image distribution (Brownlee, 2019c).
FID	Quantitative, Model-based	+	-	Sensitive for slight enhancements such as blurriness, textures, small artifacts, intra-class mode collapse, etc. Calculates the distance between the generated image and real image inception feature vectors	High bias problem: sample must be large enough - above 50K - for an optimal estimation (Borji, 2022).
LPIPS	Quantitative, Model-based	+	-	Calculates perceptual similarity between 2 images using pre-defined classification networks trained on human perceptual similarity judgments. The lowest, the most similar	Mostly efficient to compare images with distortions. We cannot compare an entire newly generated dataset to the trained one.
KID	Quantitative, Model-based	+	-	Uses Inception v3 for feature extraction and calculates maximum mean discrepancy MMD, i.e., dissimilarity, between the real and generated images	Best practice for ImageNet classes. Suffers when it comes to large variance: Unable to properly distinguish between close distributions (Zhao et al, 2021)
CrossLID	Quantitative, Manifold	+	-	Robust to small scale noise, image transformation, and sample size	Applied on simple and low dimensional data (Borji, 2022)
IMD	Quantitative, Manifold	+	-	Similar to a geometry score since it compares data distribution based on their geometry - intrinsic and multi-scale	Results are based on random approximation. Unstable IMD at each run. Otherwise, we must consider executing it multiple times and then calculate the average result or keep one IMD scoring execution for a high number of iterations (Tsitsulin, 2020).
Cross-Barcode	Quantitative, Manifold	+	-	Compares two manifolds' topology discrepancies, i.e., divergence, in high-dimensional space. Point clouds approximate manifolds. Effective in multiple domains. Faster and more sensitive than GS.	A fast computation depends on good accelerated hardware, e.g., GPU-accelerated, NVIDIA TITAN RTX (Barannikov et al, 2021)
PPL	Quantitative, Manifold	+	-	Shown a superiority over FID and P&R scores. The empirical mean of consecutive images' perceptual distance, e.g., LPIPS	Highly depends on the adopted perceptual distance function to calculate the perceptual distance of two consecutive images
D&C	Quantitative, P&R-based	-	+	Solves identical distributions match detection, outliers robustness, and the arbitrary selection of the evaluation hyperparameters issues. More interpretable and reliable evaluation than P&R	Does not distinguish synthetic data belonging to similar distributions and modes (Alaa et al, 2022)
α-P&β-R	Quantitative, P&R-based	-	+	Considers 3-dimensional evaluation metrics for precision, recall, and authenticity to quantify fidelity, diversity, and generalization for checking the generation's quality, variability coverage, and training data copying	Privacy issues since in some cases, e.g., high precision, it may copy training data with noise filters (Alaa et al, 2022)
*	Quantitative, Task-Driven	-	+	Based on the hypothesis that good quality generated data should perform as well as the trained dataset when applied in real use cases, and applications	Each application has its own of evaluating the results. It cannot be included in the automated process of a GAN model training.

could one or more classes be prone to granular collapse mode? Furthermore, it is crucial to consider the ethical implications that may lay down, such as fairness and bias generation or privacy concerns resulting in mode collapse. Therefore, researching how a training dataset

distribution would affect a GAN and its ethical implications can open a new horizon to make GANs more responsible, fair, and trustworthy.

- Despite the growing popularity of diffusion models, it's worth noting that many of the challenges and opportunities present in GAN research are still highly relevant in the current landscape. In fact, much of the foundation for diffusion generation and translation can be traced back to the pioneering work done in GANs. In contrast, diffusion models may offer advantages like increased scalability and speed. Moreover, recent breakthroughs like StyleGAN-T demonstrate that GANs are still a formidable force in the field and continue to push the boundaries of what's possible in image generation and translation. As we look to the future, it will be interesting to see how these two approaches continue to evolve and interact and what new insights and innovations will emerge. What are the limitations of diffusion models? How can GANs and diffusion models be integrated to achieve better results? How can GANs be improved to overcome the limitations of diffusion models? How can GANs be used to improve the scalability of diffusion models?

10 Conclusion

GAN has been a widely researched topic in the CV field, with applications in many broad areas. In the current era of DL and CV, GAN has gained much attention for its fast evolution and efficiency in generating adaptive images and augmenting existing datasets. In this review, we focused on GAN applications related to the industrial sector, which few publications have covered. First, we defined different GAN approaches and industrial-related applications under two main categories: image generation and domain transfer. On one side, we distinguish between conditional and unconditional synthesis for coarse and fine-grained features: we noticed that existing text-to-image synthesis models could not produce complex prompts nor generalize to cover specific industrial assets. Conversely, we highlighted texture synthesis as it is an essential concern in material and surface processing. Moreover, it is important to note that the dataset's diversity and size directly impact the performance of GANs. In cases where the dataset is small or less diverse, alternative taxonomies and architectures may be more effective. In the second part, we compared various I2I training approaches and extended the applications to practical tasks such as image expansion, recovery, and SR. Despite the interesting upscaling and sharpening results of SR, it does not generalize to barcodes, numbers, or text. In addition, we leveraged its capacity from 4x to 16x upscaling. On another side, we presented an overall discussion about the various assessment metrics. We reproduced most GAN failures and proposed a corresponding taxonomy to distinguish between GAN, neural network, adversarial learning, or synthesis model-based failures. However, relying solely on quantitative assessment for GAN generation is not practical. Finally, even with the spread of other generation technologies, GAN remains a promising area of research with numerous challenges and opportunities, particularly in synthesizing bias-free, fair, and trustworthy datasets. Overall, we hope this review serves as a basis for industrial GANs and addresses the main problem of acquiring image datasets for DL training, inspiring researchers to extend industrial AI applications.

Data availability. The datasets analyzed during the current study are self-generated and not publicly available for confidentiality reasons but are available from the corresponding author upon reasonable request.

References

Abbas A, Jain S, Gour M, et al (2021) Tomato plant disease detection using transfer learning with c-gan synthetic images. Computers and Electronics in Agriculture 187:106,279

Abou Akar C, Tekli J, Jess D, et al (2022) Synthetic object recognition dataset for industries. In: 2022 35th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), IEEE, pp 150–155

ajbrock (2019) BigGAN-PyTorch. https://github.com/ajbrock/BigGAN-PyTorch, [Online; Accessed February 08, 2022]

Alaa A, Van Breugel B, Saveliev ES, et al (2022) How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In: International Conference on Machine Learning, PMLR, pp 290–306

Alaluf Y, Patashnik O, Wu Z, et al (2022) Third time's the charm? image and video editing with stylegan3. arXiv preprint arXiv:220113433

Alami Mejjati Y, Richardt C, Tompkin J, et al (2018) Unsupervised attention-guided image-to-image translation. Advances in neural information processing systems 31

Alanov A, Kochurov M, Volkhonskiy D, et al (2019) User-controllable multi-texture synthesis with generative adversarial networks. arXiv preprint arXiv:190404751

Almahairi A, Rajeshwar S, Sordoni A, et al (2018) Augmented cyclegan: Learning many-to-many mappings from unpaired data. In: International Conference on Machine Learning, PMLR, pp 195–204

Amodio M, Krishnaswamy S (2019) Travelgan: Image-to-image translation by transformation vector learning. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition, pp 8983–8992

Anwar S, Khan S, Barnes N (2020) A deep journey into super-resolution: A survey. ACM Computing Surveys (CSUR) 53(3):1–34

Arjovsky M, Bottou L (2017) Towards principled methods for training generative adversarial networks. arXiv preprint arXiv:170104862

Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: International conference on machine learning, PMLR, pp 214–223

Armandpour M, Sadeghian A, Li C, et al (2021) Partition-guided gans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5099–5109

Arora S, Zhang Y (2017) Do gans actually learn the distribution? an empirical study. arXiv preprint arXiv:170608224

Ashok K, Boddu R, Syed SA, et al (2023) Gan base feedback analysis system for industrial iot networks. Automatika 64(2):259–267

Azulay A, Weiss Y (2018) Why do deep convolutional networks generalize so poorly to small image transformations? arXiv preprint arXiv:180512177

Bai CY, Lin HT, Raffel C, et al (2021) On training sample memorization: Lessons from benchmarking generative modeling with a large-scale competition. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, pp 2534–2542

Balakrishnan G, Zhao A, Dalca AV, et al (2018) Synthesizing images of humans in unseen poses. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8340–8348

Baldvinsson JR, Ganjalizadeh M, AlAbbasi A, et al (2022) Il-gan: Rare sample generation via incremental learning in gans. In: GLOBECOM 2022-2022 IEEE Global Communications Conference, IEEE, pp 621–626

Bansal A, Sheikh Y, Ramanan D (2017) Pixelnn: Example-based image synthesis. arXiv preprint arXiv:170805349

Bao J, Chen D, Wen F, et al (2017) Cvae-gan: fine-grained image generation through asymmetric training. In: Proceedings of the IEEE international conference on computer vision, pp 2745–2754

Barannikov S, Trofimov I, Sotnikov G, et al (2021) Manifold topology divergence: a framework for comparing data manifolds. Advances in Neural Information Processing Systems 34

Barua S, Ma X, Erfani SM, et al (2019) Quality evaluation of gans using cross local intrinsic dimensionality. arXiv preprint arXiv:190500643

Bashir SMA, Wang Y, Khan M, et al (2021) A comprehensive review of deep learning-based single image super-resolution. PeerJ Computer

Science 7:e621

Bau D, Zhu JY, Wulff J, et al (2019) Seeing what a gan cannot generate. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 4502–4511

Benaim S, Wolf L (2017) One-sided unsupervised domain mapping. Advances in neural information processing systems 30

Benaim S, Wolf L (2018) One-shot unsupervised cross domain translation. advances in neural information processing systems 31

Bergmann U, Jetchev N, Vollgraf R (2017) Learning texture manifolds with the periodic spatial gan. arXiv preprint arXiv:170506566

Bernsen NO (2008) Multimodality theory. In: Multimodal User Interfaces. Springer, p 5–29

Bhagwatkar R, Bachu S, Fitter K, et al (2020) A review of video generation approaches. In: 2020 International Conference on Power, Instrumentation, Control and Computing (PICC), IEEE, pp 1–5

Bińkowski M, Sutherland DJ, Arbel M, et al (2018) Demystifying mmd gans. arXiv preprint arXiv:180101401

Bora A, Price E, Dimakis AG (2018) Ambientgan: Generative models from lossy measurements. In: International conference on learning representations

Borji A (2022) Pros and cons of gan evaluation measures: New developments. Computer Vision and Image Understanding 215:103,329

Bougaham A, Bibal A, Linden I, et al (2021) Ganodip-gan anomaly detection through intermediate patches: a pcba manufacturing case. In: Third International Workshop on Learning with Imbalanced Domains: Theory and Applications, PMLR, pp 104–117

Boulahbal HE, Voicila A, Comport AI (2021) Are conditional GANs explicitly conditional? In: British Machine Vision Conference, Virtual, United Kingdom, URL https://hal.science/hal-03454522

Brock A, Donahue J, Simonyan K (2018) Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:180911096

Brownlee J (2019a) A Gentle Introduction to BigGAN the Big Generative Adversarial Network. https://machinelearningmastery.com/a-gentle-introduction-to-the-biggan/, [Online; Accessed February 08, 2022]

Brownlee J (2019b) How to Identify and Diagnose GAN Failure Modes. https://machinelearningmastery.com/practical-guide-to-gan-failure-modes/, [Online; Accessed May 18, 2022]

Brownlee J (2019c) How to Implement the Inception Score (IS) for Evaluating GANs. https://machinelearningmastery.com/how-to-implement-the-inception-score-from-scratch-for-evaluating-generated-images/, [Online; Accessed May 28, 2022]

Cai Y, Wang X, Yu Z, et al (2019) Dualattn-gan: Text to image synthesis with dual attentional generative adversarial network. IEEE Access 7:183,706–183,716

Cai Z, Xiong Z, Xu H, et al (2021) Generative adversarial networks: A survey toward private and secure applications. ACM Computing Surveys (CSUR) 54(6):1–38

Cao G, Zhao Y, Ni R, et al (2011) Unsharp masking sharpening detection via overshoot artifacts analysis. IEEE Signal Processing Letters 18(10):603–606

Cao J, Katzir O, Jiang P, et al (2018) Dida: Disentangled synthesis for domain adaptation. arXiv preprint arXiv:180508019

Casanova A, Careil M, Verbeek J, et al (2021) Instance-conditioned gan. Advances in Neural Information Processing Systems 34

Castillo C, De S, Han X, et al (2017) Son of zorn's lemma: Targeted style transfer using instance-aware semantic segmentation. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp 1348–1352

Chang HY, Wang Z, Chuang YY (2020) Domain-specific mappings for generative adversarial style transfer. In: European Conference on Computer Vision, Springer, pp 573–589

Chen H, Liu J, Chen W, et al (2022a) Exemplar-based pattern synthesis with implicit periodic field network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 3708–3717

Chen T, Zhang Y, Huo X, et al (2022b) Sphericgan: Semi-supervised hyper-spherical generative adversarial networks for fine-grained image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 10,001–10,010

Chen X, Jia C (2021) An overview of image-to-image translation using generative adversarial networks. In: International Conference on Pattern Recognition, Springer, pp 366–380

Chen X, Duan Y, Houthooft R, et al (2016) Infogan: Interpretable representation learning by information maximizing generative adversarial nets. Advances in neural information processing systems 29

Chen X, Xu C, Yang X, et al (2018) Attention-gan for object transfiguration in wild images. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 164–180

Choi Y, Choi M, Kim M, et al (2018) Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8789–8797

Chong MJ, Forsyth D (2020) Effectively unbiased fid and inception score and where to find them. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6070–6079

Chu C, Zhmoginov A, Sandler M (2017) Cyclegan, a master of steganography. arXiv preprint arXiv:171202950

Cohen T, Wolf L (2019) Bidirectional one-shot unsupervised domain mapping. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1784–1792

CompVis (2022) Stable diffusion model card. https://github.com/CompVis/stable-diffusion/blob/main/Stable_Diffusion_v1_Model_Card.md, [Online; Accessed January 24, 2023]

Cordts M, Omran M, Ramos S, et al (2015) The cityscapes dataset. In: CVPR Workshop on the Future of Datasets in Vision, sn

Cunningham P, Cord M, Delany SJ (2008) Supervised learning. In: Machine learning techniques for multimedia. Springer, p 21–49

Deng J, Dong W, Socher R, et al (2009) Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, Ieee, pp 248–255

Deng L (2012) The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine 29(6):141–142

Denton EL, Chintala S, Fergus R, et al (2015) Deep generative image models using a laplacian pyramid of adversarial networks. Advances in neural information processing systems 28

Denton EL, et al (2017) Unsupervised learning of disentangled representations from video. Advances in neural information processing systems 30

Ding M, Yang Z, Hong W, et al (2021) Cogview: Mastering text-to-image generation via transformers. Advances in Neural Information Processing Systems 34

Dong C, Kumar A, Liu E (2022) Think twice before detecting gan-generated fake images from their spectral domain imprints. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7865–7874

Dumoulin V, Shlens J, Kudlur M (2016) A learned representation for artistic style. arXiv preprint

arXiv:161007629

Dumoulin V, Perez E, Schucher N, et al (2018) Feature-wise transformations. Distill 3(7):e11

Durall R, Chatzimichailidis A, Labus P, et al (2020) Combating mode collapse in gan training: An empirical analysis using hessian eigenvalues. arXiv preprint arXiv:201209673

Eckerli F, Osterrieder J (2021) Generative adversarial networks in finance: an overview. arXiv preprint arXiv:210606364

Esser P, Sutter E, Ommer B (2018) A variational u-net for conditional appearance and shape generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8857–8866

Facebook Research (2021) IC-GAN: Instance-Conditioned GAN. https://github.com/facebookresearch/ic_gan, [Online; Accessed February 08, 2022]

Farajzadeh-Zanjani M, Razavi-Far R, Saif M, et al (2022) Generative adversarial networks: a survey on training, variants, and applications. In: Generative Adversarial Learning: Architectures and Applications. Springer, p 7–29

Frühstück A, Alhashim I, Wonka P (2019) Tilegan: synthesis of large-scale non-homogeneous textures. ACM Transactions on Graphics (ToG) 38(4):1–11

Fu H, Gong M, Wang C, et al (2019) Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 2427–2436

Gatys L, Ecker AS, Bethge M (2015a) Texture synthesis using convolutional neural networks. Advances in neural information processing systems 28

Gatys LA, Ecker AS, Bethge M (2015b) A neural algorithm of artistic style. arXiv preprint arXiv:150806576

Gatys LA, Bethge M, Hertzmann A, et al (2016a) Preserving color in neural artistic style transfer. arXiv preprint arXiv:160605897

Gatys LA, Ecker AS, Bethge M (2016b) Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2414–2423

Geyer J, Kassahun Y, Mahmudi M, et al (2020) A2d2: Audi autonomous driving dataset. arXiv preprint arXiv:200406320

Ghiasi G, Lee H, Kudlur M, et al (2017) Exploring the structure of a real-time, arbitrary neural artistic stylization network. arXiv preprint arXiv:170506830

Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, pp 249–256

GM H, Sahu A, Gourisaria MK (2021) Gm score: Incorporating inter-class and intra-class generator diversity, discriminability of disentangled representation, and sample fidelity for evaluating gans. arXiv preprint arXiv:211206431

Gokaslan A, Ramanujan V, Ritchie D, et al (2018) Improving shape deformation in unsupervised image-to-image translation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 649–665

Gomi T, Sakai R, Hara H, et al (2021) Usefulness of a metal artifact reduction algorithm in digital tomosynthesis using a combination of hybrid generative adversarial networks. Diagnostics 11(9):1629

Gonzalez-Garcia A, Van De Weijer J, Bengio Y (2018) Image-to-image translation for cross-domain disentanglement. Advances in neural information processing systems 31

Goodfellow I (2016) Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:170100160

Goodfellow I, Pouget-Abadie J, Mirza M, et al (2014) Generative adversarial nets. Advances in neural information processing systems 27

Google Developers (2022) GAN Training. https://developers.google.com/machine-learning/gan/training, [Online; Accessed June 05, 2022]

Gu S, Zhang R, Luo H, et al (2021) Improved sin-gan integrated with an attentional mechanism for remote sensing image classification. Remote Sensing 13(9):1713

Gulrajani I, Ahmed F, Arjovsky M, et al (2017) Improved training of wasserstein gans. Advances in neural information processing systems 30

Guo X, Wang Z, Yang Q, et al (2020) Gan-based virtual-to-real image translation for urban scene semantic segmentation. Neurocomputing 394:127–135

Gupta RK, Mahajan S, Misra R (2023) Resource orchestration in network slicing using gan-based distributional deep q-network for industrial applications. The Journal of Supercomputing 79(5):5109–5138

Härkönen E, Hertzmann A, Lehtinen J, et al (2020) Ganspace: Discovering interpretable gan controls. Advances in Neural Information Processing Systems 33:9841–9850

Hasan M, Dipto AZ, Islam MS, et al (2019) A smart semi-automated multifarious surveillance bot for outdoor security using thermal image processing. Advances in Networks 7(2):21–28

Hatanaka S, Nishi H (2021) Efficient gan-based unsupervised anomaly sound detection for refrigeration units. In: 2021 IEEE 30th International Symposium on Industrial Electronics (ISIE), IEEE, pp 1–7

He K, Zhang X, Ren S, et al (2015) Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision, pp 1026–1034

He K, Zhang X, Ren S, et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

He M, Chen D, Liao J, et al (2018) Deep exemplar-based colorization. ACM Transactions on Graphics (TOG) 37(4):1–16

Heusel M, Ramsauer H, Unterthiner T, et al (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems 30

Hindistan YS, Yetkin EF (2023) A hybrid approach with gan and dp for privacy preservation of iiot data. IEEE Access

Huang K, Wang Y, Tao M, et al (2020) Why do deep residual networks generalize better than deep feedforward networks?—a neural tangent kernel perspective. Advances in neural information processing systems 33:2698–2709

Huang X, Belongie S (2017) Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE international conference on computer vision, pp 1501–1510

Huang X, Liu MY, Belongie S, et al (2018) Multimodal unsupervised image-to-image translation. In: Proceedings of the European conference on computer vision (ECCV), pp 172–189

Huang X, Mallya A, Wang TC, et al (2022) Multimodal conditional image synthesis with product-of-experts gans. In: European Conference on Computer Vision, Springer, pp 91–109

IBM Cloud Education (2020) Supervised Learning. https://www.ibm.com/cloud/learn/supervised-learning, [Online; Accessed June 01, 2022]

Isola P, Zhu JY, Zhou T, et al (2017) Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1125–1134

Jacques S, Christe B (2020) Chapter 2 - healthcare technology basics. In: Jacques S, Christe B (eds) Introduction to Clinical Engineering. Academic Press, p 21–50, https://doi.org/https://doi.org/10.1016/B978-0-12-818103-4.00002-8, URL https://www.sciencedirect.com/science/article/pii/B9780128181034000028

Jayram T, Marois V, Kornuta T, et al (2019) Transfer learning in visual and relational reasoning. arXiv preprint arXiv:191111938

Jelinek F, Mercer RL, Bahl LR, et al (1977) Perplexity—a measure of the difficulty of speech recognition tasks. The Journal of the Acoustical Society of America 62(S1):S63–S63

Jetchev N, Bergmann U, Vollgraf R (2016) Texture synthesis with spatial generative adversarial networks. arXiv preprint arXiv:161108207

Jing Y, Yang Y, Feng Z, et al (2019) Neural style transfer: A review. IEEE transactions on visualization and computer graphics 26(11):3365–3385

Johnson DH (2006) Signal-to-noise ratio. Scholarpedia 1(12):2088

Johnson J, Alahi A, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision, Springer, pp 694–711

Joo D, Kim D, Kim J (2018) Generating a fusion image: One's identity and another's shape. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1635–1643

Junyanz (2017) PyTorch CycleGAN and Pix2Pix. https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix/blob/master/docs/datasets.md, [Online; Accessed January 25, 2023]

Karlinsky L, Shtok J, Tzur Y, et al (2017) Fine-grained recognition of thousands of object categories with single-example training. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4113–4122

Karnewar A, Wang O (2020) Msg-gan: Multi-scale gradients for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7799–7808

Karras T, Aila T, Laine S, et al (2017) Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:171010196

Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4401–4410

Karras T, Aittala M, Hellsten J, et al (2020a) Training generative adversarial networks with limited data. Advances in Neural Information Processing Systems 33:12,104–12,114

Karras T, Laine S, Aittala M, et al (2020b) Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 8110–8119

Karras T, Aittala M, Laine S, et al (2021) Alias-free generative adversarial networks. Advances in Neural Information Processing Systems 34

Kaymak Ç, Uçar A (2019) A brief survey and an application of semantic image segmentation for autonomous driving. In: Handbook of Deep Learning Applications. Springer, p 161–200

Kazemi H, Soleymani S, Taherkhani F, et al (2018) Unsupervised image-to-image translation using domain-specific variational information bound. Advances in neural information processing systems 31

Khrulkov V, Oseledets I (2018) Geometry score: A method for comparing generative adversarial networks. In: International Conference on Machine Learning, PMLR, pp 2621–2629

Kilgour K, Zuluaga M, Roblek D, et al (2019) Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. In: INTERSPEECH, pp 2350–2354

Kim DW, Ryun Chung J, Jung SW (2019a) Grdn: Grouped residual dense network for real image denoising and gan-based real-world noise modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops

Kim H, Mnih A (2018) Disentangling by factorising. In: International Conference on Machine Learning, PMLR, pp 2649–2658

Kim J, Park H (2022) Limited discriminator gan using explainable ai model for overfitting problem. ICT Express

Kim J, Kim M, Kang H, et al (2019b) U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. arXiv preprint arXiv:190710830

Kim T, Cha M, Kim H, et al (2017) Learning to discover cross-domain relations with generative adversarial networks. In: International conference on machine learning, PMLR, pp 1857–1865

kligvasser (2021) SinGAN. https://github.com/kligvasser/SinGAN, [Online; Accessed March 29, 2022]

Koshino K, Werner RA, Pomper MG, et al (2021) Narrative review of generative adversarial networks in medical and molecular imaging. Annals of Translational Medicine 9(9)

Kupyn O, Martyniuk T, Wu J, et al (2019) Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 8878–8887

Ledig C, Theis L, Huszár F, et al (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690

Lee HY, Tseng HY, Huang JB, et al (2018) Diverse image-to-image translation via disentangled representations. In: Proceedings of the European conference on computer vision (ECCV), pp 35–51

Li B, Zou Y, Zhu R, et al (2022) Fabric defect segmentation system based on a lightweight gan for industrial internet of things. Wireless Communications and Mobile Computing 2022

Li C, Wand M (2016) Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European conference on computer vision, Springer, pp 702–716

Li K, Yang S, Dong R, et al (2020a) Survey of single image super-resolution reconstruction. IET Image Processing 14(11):2273–2290

Li M, Huang H, Ma L, et al (2018) Unsupervised image-to-image translation with stacked cycle-consistent adversarial networks. In: Proceedings of the European conference on computer vision (ECCV), pp 184–199

Li M, Ye C, Li W (2019) High-resolution network for photorealistic style transfer. arXiv preprint arXiv:190411617

Li R, Cao W, Jiao Q, et al (2020b) Simplified unsupervised image translation for semantic segmentation adaptation. Pattern Recognition 105:107,343

Li Y, Fang C, Yang J, et al (2017) Diversified texture synthesis with feed-forward networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3920–3928

Li Y, Singh KK, Ojha U, et al (2020c) Mixnmatch: Multifactor disentanglement and encoding for conditional image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8039–8048

Li Y, Zhang R, Lu J, et al (2020d) Few-shot image generation with elastic weight consolidation. arXiv preprint arXiv:201202780

Li Y, Sixou B, Peyrin F (2021) A review of the deep learning methods for medical images super resolution problems. IRBM 42(2):120–133

Likas A, Vlassis N, Verbeek JJ (2003) The global k-means clustering algorithm. Pattern recognition 36(2):451–461

Lin J, Xia Y, Qin T, et al (2018) Conditional image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5524–5532

Lin J, Wang Y, He T, et al (2019) Learning to transfer: Unsupervised meta domain translation. arXiv preprint arXiv:190600181

Lin J, Pang Y, Xia Y, et al (2020) Tuigan: Learning versatile image-to-image translation with two unpaired images. In: European Conference on Computer Vision, Springer, pp 18–35

Lin TY, Maire M, Belongie S, et al (2014) Microsoft coco: Common objects in context. In: European conference on computer vision, Springer, pp 740–755

Liu G, Taori R, Wang TC, et al (2020) Transposer: Universal texture synthesis using feature maps as transposed convolution filter. arXiv preprint arXiv:200707243

Liu H, Cao S, Ling Y, et al (2021) Inpainting for saturation artifacts in optical coherence tomography using dictionary-based sparse representation. IEEE photonics journal 13(2)

Liu MY, Breuel T, Kautz J (2017) Unsupervised image-to-image translation networks. Advances in neural information processing systems 30

Liu Z, Liu C, Shum HY, et al (2002) Pattern-based texture metamorphosis. In: 10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings., IEEE, pp 184–191

Liu Z, Li M, Zhang Y, et al (2023) Fine-grained face swapping via regional gan inversion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8578–8587

Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440

Lorenz D, Bereska L, Milbich T, et al (2019) Unsupervised part-based disentangling of object shape and appearance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 10,955–10,964

Luan F, Paris S, Shechtman E, et al (2017) Deep photo style transfer. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4990–4998

Luhman T, Luhman E (2023) High fidelity image synthesis with deep vaes in latent space. arXiv preprint arXiv:230313714

Ma L, Jia X, Georgoulis S, et al (2018a) Exemplar guided unsupervised image-to-image translation with semantic consistency. arXiv preprint arXiv:180511145

Ma L, Sun Q, Georgoulis S, et al (2018b) Disentangled person image generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 99–108

Mao Q, Lee HY, Tseng HY, et al (2019) Mode seeking generative adversarial networks for diverse image synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1429–1437

Mao X, Li Q, Xie H, et al (2017) Least squares generative adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2794–2802

Maqsood S, Javed U (2020) Multi-modal medical image fusion based on two-scale image decomposition and sparse representation. Biomedical Signal Processing and Control 57:101,810

Mathiasen A, Hvilshøj F (2020) Backpropagating through fr\'echet inception distance. arXiv preprint arXiv:200914075

McCloskey S, Albright M (2019) Detecting gan-generated imagery using saturation cues. In: 2019 IEEE international conference on image processing (ICIP), IEEE, pp 4584–4588

Meta AI (2021) Building AI that can generate images of things it has never seen before. https://ai.facebook.com/blog/instance-conditioned-gans/, [Online; Accessed February

09, 2022]

Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv preprint arXiv:14111784

Mittal T (2019) Tips On Training Your GANs Faster and Achieve Better Results. https://medium.com/intel-student-ambassadors/tips-on-training-your-gans-faster-and-achieve-better-results-9200354acaa5, [Online; Accessed May 18, 2022]

Miyato T, Kataoka T, Koyama M, et al (2018) Spectral normalization for generative adversarial networks. arXiv preprint arXiv:180205957

Mo S, Cho M, Shin J (2018) Instagan: Instance-aware image-to-image translation. arXiv preprint arXiv:181210889

Mo S, Cho M, Shin J (2020) Freeze the discriminator: a simple baseline for fine-tuning gans. arXiv preprint arXiv:200210964

Mordvintsev A, Olah C, Tyka M (2015) Inceptionism: Going deeper into neural networks, 2015. URL https://research google-blog com/2015/06/inceptionism-going-deeper-into-neural html

Murez Z, Kolouri S, Kriegman D, et al (2018) Image to image translation for domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4500–4509

Mustafa A, Mantiuk RK (2020) Transformation consistency regularization–a semi-supervised paradigm for image-to-image translation. In: European Conference on Computer Vision, Springer, pp 599–615

Naeem MF, Oh SJ, Uh Y, et al (2020) Reliable fidelity and diversity metrics for generative models. In: International Conference on Machine Learning, PMLR, pp 7176–7185

Nakano R (2018) Arbitrary style transfer in TensorFlow.js. https://magenta.tensorflow.org/blog/2018/12/20/style-transfer-js/, [Online; Accessed April 04, 2022]

Nash C, Menick J, Dieleman S, et al (2021) Generating images with sparse representations. arXiv preprint arXiv:210303841

Naumann A, Hertlein F, Doerr L, et al (2023) Literature review: Computer vision applications in transportation logistics and warehousing. arXiv preprint arXiv:230406009

Nedeljković D, Jakovljević Ž (2022) Gan-based data augmentation in the design of cyber-attack detection methods. In: 9th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2022), Proceedings, Novi Pazar, June 2022, ROI1. 4, ETRAN Society, Belgrade, Academic Mind, Belgrade, pp ROI1–4

Nie W, Karras T, Garg A, et al (2020) Semi-supervised stylegan for disentanglement learning. In: International Conference on Machine Learning, PMLR, pp 7360–7369

Nielsen M (2019) Deep Learning - Chapter 6. http://neuralnetworksanddeeplearning.com/chap6.html, [Online; Accessed February 04, 2022]

Noguchi A, Harada T (2019) Image generation from small datasets via batch statistics adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2750–2758

NVLabs (2020) StyleGAN2 with adaptive discriminator augmentation (ADA) — Official TensorFlow implementation. https://github.com/NVlabs/stylegan2-ada, [Online; Accessed February 07, 2022]

NVLabs (2021) Official PyTorch implementation of the NeurIPS 2021 paper. https://github.com/NVlabs/stylegan3, [Online; Accessed February 08, 2022]

NVLabs (2021a) StyleGAN - Official TensorFlow Implementation. https://github.com/NVlabs/stylegan, [Online; Accessed February 03, 2022]

NVLabs (2021b) StyleGAN2 - Official TensorFlow Implementation. https://github.com/NVlabs/stylegan2, [Online; Accessed February 03, 2022]

NVLabs (2021) StyleGAN2-ADA — Official PyTorch implementation. https://github.com/NVlabs/stylegan2-ada-pytorch, [Online; Accessed February 08, 2022]

Odena A, Dumoulin V, Olah C (2016) Deconvolution and Checkerboard Artifacts. https://distill.pub/2016/deconv-checkerboard/, [Online; Accessed June 03, 2022]

Open AI (2022) CLIP: Connecting Text and Images. https://openai.com/blog/clip/, [Online; Accessed February 14, 2022]

openai (2022) CLIP. https://github.com/openai/CLIP, [Online; Accessed February 14, 2022]

Pan SJ, Yang Q (2009) A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22(10):1345–1359

Pan X, Tewari A, Leimkühler T, et al (2023) Drag your gan: Interactive point-based manipulation on the generative image manifold. arXiv preprint arXiv:230510973

Pang Y, Lin J, Qin T, et al (2021) Image-to-image translation: Methods and applications. IEEE Transactions on Multimedia

Park SJ, Son H, Cho S, et al (2018) Srfeat: Single image super-resolution with feature discrimination. In: Proceedings of the European conference on computer vision (ECCV), pp 439–455

Park T, Liu MY, Wang TC, et al (2019) Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2337–2346

Park T, Efros AA, Zhang R, et al (2020a) Contrastive learning for unpaired image-to-image translation. In: European Conference on Computer Vision, Springer, pp 319–345

Park T, Zhu JY, Wang O, et al (2020b) Swapping autoencoder for deep image manipulation. Advances in Neural Information Processing Systems 33:7198–7211

Parmar G, Zhang R, Zhu JY (2021) On buggy resizing libraries and surprising subtleties in fid calculation. arXiv preprint arXiv:210411222

Pasini M (2019) 10 Lessons I Learned Training GANs for one Year. https://towardsdatascience.com/10-lessons-i-learned-training-generative-adversarial-networks-gans-for-a-year-c9071159628, [Online; Accessed May 18, 2022]

Pasquini C, Laiti F, Lobba D, et al (2023) Identifying synthetic faces through gan inversion and biometric traits analysis. Applied Sciences 13(2):816

Patashnik O, Wu Z, Shechtman E, et al (2021) Styleclip: Text-driven manipulation of stylegan imagery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2085–2094

Pathak D, Krahenbuhl P, Donahue J, et al (2016) Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2536–2544

Pavan Kumar M, Jayagopal P (2021) Generative adversarial networks: a survey on applications and challenges. International Journal of Multimedia Information Retrieval 10(1):1–24

Paysan P, Knothe R, Amberg B, et al (2009) A 3d face model for pose and illumination invariant face recognition. In: 2009 sixth IEEE international conference on advanced video and signal based surveillance, Ieee, pp 296–301

Peng X, Yu X, Sohn K, et al (2017) Reconstruction-based disentanglement for pose-invariant face recognition. In: Proceedings of the IEEE international conference on computer vision, pp 1623–1632

Petrovic V, Cootes T (2006) Information representation for image fusion evaluation. In: 2006 9th International Conference on Information Fusion, IEEE, pp 1–7

Portenier T, Arjomand Bigdeli S, Goksel O (2020) Gramgan: Deep 3d texture synthesis from 2d

exemplars. Advances in Neural Information Processing Systems 33:6994–7004

Preuer K, Renz P, Unterthiner T, et al (2018) Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. Journal of chemical information and modeling 58(9):1736–1741

Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:151106434

Radford A, Kim JW, Hallacy C, et al (2021) Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, PMLR, pp 8748–8763

Ramesh A, Pavlov M, Goh G, et al (2021) Zero-shot text-to-image generation. In: International Conference on Machine Learning, PMLR, pp 8821–8831

Ramesh A, Dhariwal P, Nichol A, et al (2022) Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:220406125

Ravuri S, Vinyals O (2019a) Classification accuracy score for conditional generative models. Advances in neural information processing systems 32

Ravuri S, Vinyals O (2019b) Seeing is not necessarily believing: Limitations of biggans for data augmentation

Richter SR, Vineet V, Roth S, et al (2016) Playing for data: Ground truth from computer games. In: European conference on computer vision, Springer, pp 102–118

Roich D, Mokady R, Bermano AH, et al (2022) Pivotal tuning for latent-based editing of real images. ACM Transactions on Graphics (TOG) 42(1):1–13

Rombach R, Blattmann A, Lorenz D, et al (2022) High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 10,684–10,695

Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention, Springer, pp 234–241

Rutinowski J, Youssef H, Gouda A, et al (2022) The potential of deep learning based computer vision in warehousing logistics. Logistics Journal: Proceedings 2022(18)

Saad MM, Rehmani MH, O'Reilly R (2022) Addressing the intra-class mode collapse problem using adaptive input image normalization in gan-based x-ray images. arXiv preprint arXiv:220110324

Sajjadi MS, Bachem O, Lucic M, et al (2018) Assessing generative models via precision and recall. Advances in Neural Information Processing Systems 31

Salimans T, Goodfellow I, Zaremba W, et al (2016) Improved techniques for training gans. Advances in neural information processing systems 29

Sauer A, Schwarz K, Geiger A (2022) Stylegan-xl: Scaling stylegan to large diverse datasets. In: ACM SIGGRAPH 2022 conference proceedings, pp 1–10

Sauer A, Karras T, Laine S, et al (2023) Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. arXivorg abs/2301.09515. URL https://arxiv.org/abs/2301.09515

Saunshi N, Ash J, Goel S, et al (2022) Understanding contrastive learning requires incorporating inductive biases. arXiv preprint arXiv:220214037

Schuh G, Anderl R, Gausemeier J, et al (2017) Industrie 4.0 Maturity Index: Die digitale Transformation von Unternehmen gestalten. Herbert Utz Verlag

Shaham TR, Dekel T, Michaeli T (2019) Singan: Learning a generative model from a single natural image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 4570–4580

Shahbazi M, Danelljan M, Paudel DP, et al (2022) Collapse by conditioning: Training class-conditional gans with limited data. arXiv preprint arXiv:220106578

Shannon CE (1949) Communication in the presence of noise. Proceedings of the IRE 37(1):10–21

Sharma M, Verma A, Vig L (2019) Learning to clean: A gan perspective. In: Computer Vision–ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers 14, Springer, pp 174–185

Shen Y, Zhou B (2021) Closed-form factorization of latent semantics in gans. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1532–1540

Shen Z, Huang M, Shi J, et al (2019) Towards instance-level image-to-image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 3683–3692

Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. Journal of big data 6(1):1–48

Singh KK, Ojha U, Lee YJ (2019) Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6490–6499

Song Q, Li G, Wu S, et al (2023) Discriminator feature-based progressive gan inversion. Knowledge-Based Systems 261:110,186

Song S, Yu F, Zeng A, et al (2017) Semantic scene completion from a single depth image. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1746–1754

Song Y, Yang C, Lin Z, et al (2018) Contextual-based image inpainting: Infer, match, and translate. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 3–19

SORDI.ai (2023) Synthetic Object Recognition Dataset for Industries. https://www.sordi.ai, [Online; Accessed May 24, 2023]

Soucy P, Mineau GW (2001) A simple knn algorithm for text categorization. In: Proceedings 2001 IEEE international conference on data mining, IEEE, pp 647–648

Struski Ł, Knop S, Spurek P, et al (2022) Locogan—locally convolutional gan. Computer Vision and Image Understanding 221:103,462

Suárez PL, Sappa AD, Vintimilla BX (2017) Infrared image colorization based on a triplet dcgan architecture. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 18–23

Sussillo D, Abbott L (2014) Random walk initialization for training very deep feedforward networks. arXiv preprint arXiv:14126558

Suzuki R, Koyama M, Miyato T, et al (2018) Spatially controllable image synthesis with internal representation collaging. arXiv preprint arXiv:181110153

taesungp (2020) Contrastive Unpaired Translation (CUT). https://github.com/taesungp/contrastive-unpaired-translation, [Online; Accessed February 07, 2022]

tamarott (2020) SinGAN. https://github.com/tamarott/SinGAN, [Online; Accessed March 29, 2022]

Tancik M, Srinivasan P, Mildenhall B, et al (2020) Fourier features let networks learn high frequency functions in low dimensional domains. Advances in Neural Information Processing Systems 33:7537–7547

Tang CS, Veelenturf LP (2019) The strategic role of logistics in the industry 4.0 era. Transportation Research Part E: Logistics and Transportation Review 129:1–11

Tang H, Xu D, Sebe N, et al (2019) Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2417–2426

Tang H, Xu D, Yan Y, et al (2020) Local class-specific and global image-level generative adversarial networks for semantic-guided scene generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7870–7879

Tao M, Tang H, Wu F, et al (2022) Df-gan: A simple and effective baseline for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 16,515–16,525

Tao S, Wang J (2020) Alleviation of gradient exploding in gans: Fake can be real. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1191–1200

Teterwak P, Sarna A, Krishnan D, et al (2019) Boundless: Generative adversarial networks for image extension. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 10,521–10,530

Tian C, Fei L, Zheng W, et al (2020) Deep learning on image denoising: An overview. Neural Networks 131:251–275

tkarras (2017a) Progressive Growing of GANs for Improved Quality, Stability, and Variation — Official TensorFlow implementation of the ICLR 2018 paper. https://github.com/tkarras/progressive_growing_of_gans, [Online; Accessed February 07, 2022]

tkarras (2017b) Progressive Growing of GANs for Improved Quality, Stability, and Variation — Official Theano implementation of the ICLR 2018 paper. https://github.com/tkarras/progressive_growing_of_gans/tree/original-theano-version, [Online; Accessed February 07, 2022]

Torrey L, Shavlik J (2010) Transfer learning. In: Handbook of research on machine learning applications and trends: algorithms, methods, and techniques. IGI global, p 242–264

tportenier (2020) GramGAN: Deep 3D Texture Synthesis From 2D Exemplars. https://github.com/tportenier/gramgan, [Online; Accessed March 29, 2022]

Tran LD, Nguyen SM, Arai M (2020) Gan-based noise model for denoising real images. In: Proceedings of the Asian Conference on Computer Vision

Tran NT, Tran VH, Nguyen NB, et al (2021) On data augmentation for gan training. IEEE Transactions on Image Processing 30:1882–1897

Tsitsulin A (2020) Different results on the same arrays. https://github.com/xgfs/imd/issues/2, [Online; Accessed May 28, 2022]

Tsitsulin A, Munkhoeva M, Mottin D, et al (2019) The shape of data: Intrinsic distance for data distributions. arXiv preprint arXiv:190511141

Tzovaras D (2008) Multimodal user interfaces: from signals to interaction. Springer

Ulyanov D, Lebedev V, Vedaldi A, et al (2016a) Texture networks: Feed-forward synthesis of textures and stylized images. In: ICML, p 4

Ulyanov D, Vedaldi A, Lempitsky V (2016b) Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:160708022

Vo DM, Sugimoto A, Nakayama H (2022) Ppcd-gan: Progressive pruning and class-aware distillation for large-scale conditional gans compression. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 2436–2444

Voita L (2022) (Introduction to) Transfer Learning. https://lena-voita.github.io/nlp_course/transfer_learning.html, [Online; Accessed June 03, 2022]

Wang M, Lang C, Liang L, et al (2021a) Fine-grained semantic image synthesis with object-attention generative adversarial network. ACM

Transactions on Intelligent Systems and Technology (TIST) 12(5):1–18

Wang TC, Liu MY, Zhu JY, et al (2018a) High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8798–8807

Wang X, Yu K, Wu S, et al (2018b) Esrgan: Enhanced super-resolution generative adversarial networks. In: Proceedings of the European conference on computer vision (ECCV) workshops

Wang X, Xie L, Dong C, et al (2021b) Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1905–1914

Wang Y, Qian B, Li B, et al (2013) Metal artifacts reduction using monochromatic images from spectral ct: evaluation of pedicle screws in patients with scoliosis. European journal of radiology 82(8):e360–e366

Wang Y, Wu C, Herranz L, et al (2018c) Transferring gans: generating images from limited data. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 218–234

Wang Y, Tao X, Shen X, et al (2019) Wide-context semantic image extrapolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1399–1408

Wang Y, Gonzalez-Garcia A, Berga D, et al (2020a) Minegan: effective knowledge transfer from gans to target domains with few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9332–9341

Wang Y, Yao Q, Kwok JT, et al (2020b) Generalizing from a few examples: A survey on few-shot learning. ACM computing surveys (csur) 53(3):1–34

Wang Z, Bovik AC, Sheikh HR, et al (2004) Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing 13(4):600–612

Wang Z, Chen J, Hoi SC (2020c) Deep learning for image super-resolution: A survey. IEEE transactions on pattern analysis and machine intelligence 43(10):3365–3387

Wang Z, She Q, Ward TE (2021c) Generative adversarial networks in computer vision: A survey and taxonomy. ACM Computing Surveys (CSUR) 54(2):1–38

Wei LY, Lefebvre S, Kwatra V, et al (2009) State of the art in example-based texture synthesis. Eurographics 2009, State of the Art Report, EG-STAR pp 93–117

Weiss K, Khoshgoftaar TM, Wang D (2016) A survey of transfer learning. Journal of Big data 3(1):1–40

Williams L (1983) Pyramidal parametrics. In: Proceedings of the 10th annual conference on Computer graphics and interactive techniques, pp 1–11

Wu W, Cao K, Li C, et al (2019) Transgaga: Geometry-aware unsupervised image-to-image translation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 8012–8021

Xia W, Zhang Y, Yang Y, et al (2022) Gan inversion: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence

Xiao F, Liu H, Lee YJ (2019) Identity from here, pose from there: Self-supervised disentanglement and generation of objects using unlabeled videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 7013–7022

xinntao (2021) Real-ESRGAN. https://github.com/xinntao/Real-ESRGAN, [Online; Accessed May 09, 2022]

Xu T, Zhang P, Huang Q, et al (2018) Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In:

Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1316–1324

Xuan J, Yang Y, Yang Z, et al (2019) On the anomalous generalization of gans. arXiv preprint arXiv:190912638

Ye H, Yang X, Takac M, et al (2021) Improving text-to-image synthesis using contrastive learning. The 32nd British Machine Vision Conference (BMVC)

Yi X, Walia E, Babyn P (2019) Generative adversarial network in medical imaging: A review. Medical image analysis 58:101,552

Yi Z, Zhang H, Tan P, et al (2017) Dualgan: Unsupervised dual learning for image-to-image translation. In: Proceedings of the IEEE international conference on computer vision, pp 2849–2857

Yinka-Banjo C, Ugot OA (2020) A review of generative adversarial networks and its application in cybersecurity. Artificial Intelligence Review 53(3):1721–1736

Yu N, Barnes C, Shechtman E, et al (2019) Texture mixer: A network for controllable synthesis and interpolation of texture. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 12,164–12,173

Yuan Y, Liu S, Zhang J, et al (2018) Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 701–710

Yuheng-Li (2020) MixNMatch: Multifactor Disentanglement and Encoding for Conditional Image Generation. https://github.com/Yuheng-Li/MixNMatch, [Online; Accessed February 14, 2022]

Zhang H, Xu T, Li H, et al (2017a) Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 5907–5915

Zhang H, Xu T, Li H, et al (2018a) Stackgan++: Realistic image synthesis with stacked generative adversarial networks. IEEE transactions on pattern analysis and machine intelligence 41(8):1947–1962

Zhang K, Luo W, Zhong Y, et al (2020a) Deblurring by realistic blurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 2737–2746

Zhang N, Zhang L, Cheng Z (2017b) Towards simulating foggy and hazy images and evaluating their authenticity. In: International Conference on Neural Information Processing, Springer, pp 405–415

Zhang P, Zhang B, Chen D, et al (2020b) Cross-domain correspondence learning for exemplar-based image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5143–5153

Zhang Q, Liu Y, Blum RS, et al (2018b) Sparse representation based multi-sensor image fusion for multi-focus and multi-modality images: A review. Information Fusion 40:57–75

Zhang R (2019) Making convolutional networks shift-invariant again. In: International conference on machine learning, PMLR, pp 7324–7334

Zhang R, Zhu JY, Isola P, et al (2017c) Real-time user-guided image colorization with learned deep priors. arXiv preprint arXiv:170502999

Zhang R, Isola P, Efros AA, et al (2018c) The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 586–595

Zhang S, Zhen A, Stevenson RL (2019a) Gan based image deblurring using dark channel prior. arXiv preprint arXiv:190300107

Zhang X, Karaman S, Chang SF (2019b) Detecting and simulating artifacts in gan fake images. In: 2019 IEEE International Workshop on Information Forensics and Security (WIFS), IEEE, pp 1–6

Zhang Y, Liu S, Dong C, et al (2019c) Multiple cycle-in-cycle generative adversarial networks for unsupervised image super-resolution. IEEE transactions on Image Processing 29:1101–1112

Zhao L, Mo Q, Lin S, et al (2020a) Uctgan: Diverse image inpainting based on unsupervised cross-space translation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5741–5750

Zhao S, Liu Z, Lin J, et al (2020b) Differentiable augmentation for data-efficient gan training. Advances in Neural Information Processing Systems 33:7559–7570

Zhao S, Cui J, Sheng Y, et al (2021) Large scale image completion via co-modulated generative adversarial networks. arXiv preprint arXiv:210310428

Zhao Y, Wu R, Dong H (2020c) Unpaired image-to-image translation using adversarial consistency loss. In: European Conference on Computer Vision, Springer, pp 800–815

Zhao Z, Zhang Z, Chen T, et al (2020d) Image augmentations for gan training. arXiv preprint arXiv:200602595

Zheng C, Cham TJ, Cai J (2021) The spatially-correlative loss for various image translation tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 16,407–16,417

Zheng H, Fu J, Zha ZJ, et al (2019) Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5012–5021

Zhou X, Zhang B, Zhang T, et al (2021) Cocosnet v2: Full-resolution correspondence learning for image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 11,465–11,475

Zhou Y, Zhu Z, Bai X, et al (2018) Non-stationary texture synthesis by adversarial expansion. arXiv preprint arXiv:180504487

Zhu JY, Park T, Isola P, et al (2017a) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2223–2232

Zhu JY, Zhang R, Pathak D, et al (2017b) Toward multimodal image-to-image translation. Advances in neural information processing systems 30

Zhu M, Pan P, Chen W, et al (2019) Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5802–5810

Zhu P, Abdal R, Qin Y, et al (2020) Sean: Image synthesis with semantic region-adaptive normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5104–5113

Appendix A Experimentation Datasets

In this section, we describe our 4 in-house rendered synthetic datasets that we used in our experimentation.

Industrial Assets: This dataset consists of simple synthetic images (512 × 512 px) for different industrial assets with domain randomization. We consider the following assets combinations: smart transport robot (STR) (16,000), trolly (16,000), STR and trolly (11,457), pallet (11,271), jack (9,425), electrical jack (8,944), stillage (8,933), forklift (8,840), tugger train (8,794), small load carrier (KLT) box (9,976), and random combinations of grouped assets (4,029). This dataset was rendered using Unity Engine

Fig. A1 Sample of the Industrial Assets dataset including STR, trolly, pallet, jack, electrical jack, stillage, forklift, tugger train, KLT, and random combinations of assets.

All three remaining datasets are rendered using NVIDIA Omniverse.

KLT & Pallet: The dataset includes 4 combinations of small load carrier (KLT) box and pallet single images: (1) Low variation of KLT boxes (9,948) (2) Low variations of Pallet (10,893) (3) Higher variations of Pallet (5,000) (4) Higher variations of KLT box (2,500).

Stillage Modalities: The dataset consists of

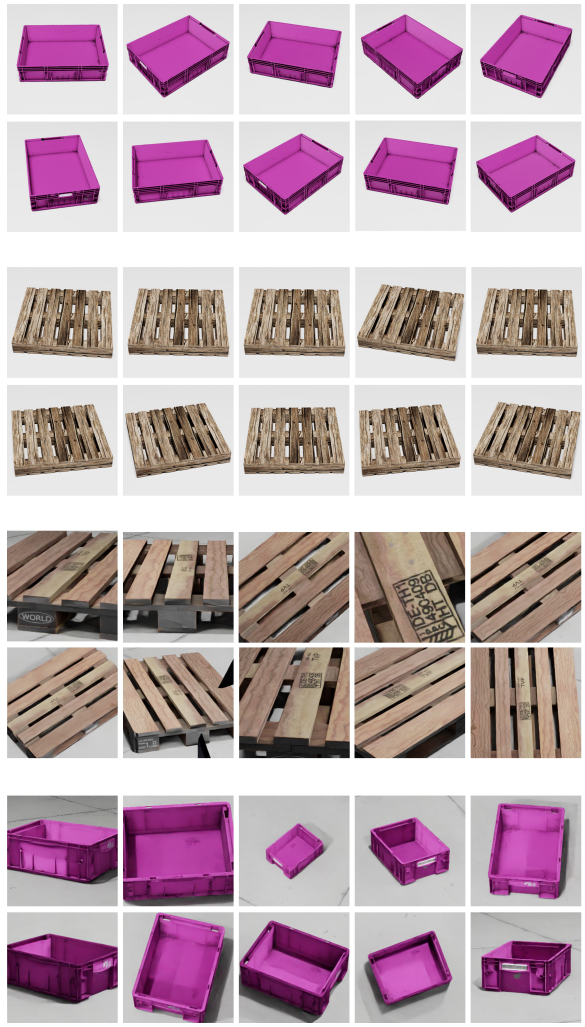


Fig. A2 Sample of the KLT & Pallet dataset. From top to bottom: low variation of KLT and Pallet, higher variation of Pallet and KLT

2,006 synthetic images (1,280 × 720 px) for different stillages, sided next to each other. In addition, paired semantic and instance segmentations, and depth images are included (In total, 8,024 images).

Klt2Cardboard: The dataset consists of around 20K synthetic images (3,206 × 1,440 px) in total for randomly stacked boxes placed on a euro pallet in two different room environments. The first 10,138 consists of small load carrier (KLT) boxes surrounded by logistic assets, while the second 10,222 images consist of cardboard boxes

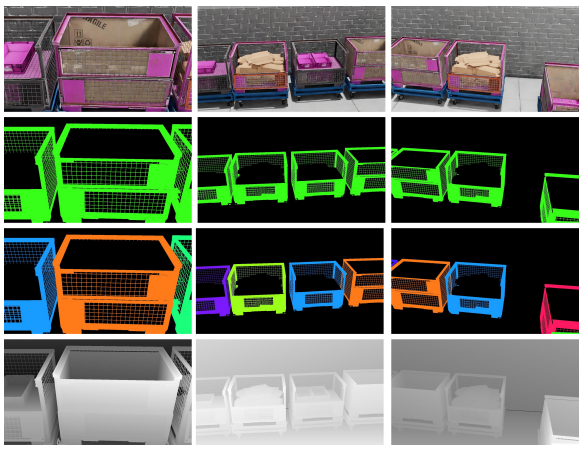


Fig. A3 Sample of the Stillage Modalities dataset. From top to bottom: RGB plain color, semantic segmentation, instance segmentation, depth images.

surrounded by office assets. Both contain a maximum of 2-sided, 5-stacked boxes.

Fig. A4 Sample of the Klt2Cardboard dataset. Top: KLT boxes, bottom: cardboard boxes

Appendix B Conditional GAN (cGAN)

It is the conditional version of GAN (Mirza and Osindero, 2014). In cGAN, conditional settings are applied to both the generator and the discriminator. The conditional settings can be any type of auxiliary information such as class labels (Karras et al, 2020a), instance images (Casanova et al, 2021), data pairing (Isola et al, 2017), etc. Alongside the latent space, the generator network inputs the class information condition and produces images. Fundamentally, a generator is free to generate whatever output as long as it satisfies the discriminator, which explains the necessity of applying conditionality on both GAN networks (Boulahbal et al, 2021). Additionally, a cGAN converges faster than a classical GAN, and its generated image random distribution follows a certain pattern.

As previously mentioned, a default simple GAN consists of 2 networks: a generative and a discriminator network. Although, training GAN's networks does not differ from training any other network. Therefore, we refer to the universal taxonomy in Machine Learning (ML) to consider four major GAN learning methods based on the different ways of handling the available training datasets: supervised, unsupervised, semi-supervised, and few-shot as stated below in Appendix C.

Appendix C GAN Training Taxonomy

Supverised learning (SL): It uses labeled datasets to train GANs (Cunningham et al, 2008). Due to the various GAN applications, we consider different shapes of labels. For instance, GAN's conditional generation requires additional category labels for each training image as in (Karras et al, 2020a, 2021; Casanova et al, 2021). Supervised I2I applications require paired image dataset to obtain accurate domain translations as in (Wang et al, 2018a; Zhu et al, 2017b). However, despite the high efficiency of supervised learning over other learning methods, ensuring labeled data needs a certain level of expertise to structure that data, is intensively time-consuming, and has a likelihood of human error (IBM Cloud Education, 2020). In addition, it is sometimes impossible to acquire, especially when it comes to paired datasets, since it is "double the trouble" as capturing and processing a single domain dataset. In this case, synchronized hardware or an entire area control are additionally required.

Unsupverised learning (UL): Unlike SL, the model is not provided with labels or domain pairings. Therefore, the model works on its own to discover patterns and information (Chu et al, 2017; Park et al, 2020a). Yet, for optimal performance, unsupervised learning requires a more extensive dataset. In fact, unlabeled datasets are much more easily accessible. Therefore, researchers have focused more on enhancing unsupervised learning architecture over supervised learning methods (Chen and Jia, 2021).

Semi-supervised learning (SSL): It lies between supervised and unsupervised learning. Typically, a semi-supervised learning operates a dataset with few labels. Therefore, it aims to label all remaining images based on the small number of existing labeled images (e.g., it is possible to reconstruct the whole video by labeling 10-20% of its frames) (Mustafa and Mantiuk, 2020). Moreover, SSL is efficient for good disentanglement learning using only 0.25-2.5% of labeled data (Nie et al, 2020). However, semi-supervised learning generally has lower accuracy since all future predictions rely on predicted GT labels.

Few-shot learning (FSL): FSL models are based on a very few (Benaim and Wolf, 2018; Cohen and Wolf, 2019) to single image, i.e. one-shot learning (OSL), training dataset (Park et al, 2020a; Lin et al, 2020; Shaham et al, 2019). By definition, *"FSL is a type of machine learning problems (specified by E, T and P), where E contains only a limited number of examples with supervised information for the target T."* Afterward, FSL training is evaluated based on its performance P (Wang et al, 2020b). Despite the fast FSL training generalization on a new small dataset, FSL models outperform in their field of experience. They may not be optimal when inferring slightly different use cases (check Section 6.3). Although, FSL is a promising field to relieve the burden of collecting huge datasets for the methods above, especially since limited hardware is enough for training. Thus, many researchers are focusing on fulfilling that aim.

Additionally, transfer learning (TL) is becoming the de facto solution for CV training (Jayram et al, 2019). The main idea is transferring knowledge from an auxiliary model into the main one. In other terms, an auxiliary model is a model that is trained on huge datasets to satisfy a source task. Afterward, the training *"is resumed"* with a smaller dataset to solve the interesting target task. Therefore, we divide TL methods into two categories: transductive and inductive. A transductive TL maintains the same task and labels as in the source task, e.g., Domain Adaptation (Guo et al, 2020; Li et al, 2020b; Cao et al, 2018; Murez et al, 2018). However, in inductive TL, the task, and therefore the labels, are changed and defined in the target task (e.g., sequential transfer learning, which is the most popular TL method) (Voita, 2022). Yet, some limitations may occur when training the model too long (check Section 4.1.2).

Appendix D StyleGAN Retrospective

D.1 StyleGAN

In 2019, NVIDIA published StyleGAN (Karras et al, 2019; NVLabs, 2021a), an extension of the traditional ProGAN architecture (Karras et al, 2017). The generator network has been modified to include progressive resolution blocks, starting from $2^2 \times 2^2$ to $2^{10} \times 2^{10}$ pixels. At each block, and after each convolution layer (Gatys et al, 2015b), a different sample of Gaussian noise is added to the feature map (Nielsen, 2019). Inspired by the style transfer literature (Huang and Belongie, 2017; Jing et al, 2019), an AdaIN layer (Huang and Belongie, 2017; Dumoulin et al, 2016, 2018; Ghiasi et al, 2017) controls the style transfer process. While the style only affects global effects, such as shape, identity, pose, lighting, and background, the noise injection at each block directly controls the image features and guarantees stochastic variations at different scales. This process separates the high-level attributes from the stochastic variation of local effects, such as beard, freckles, and hair. As a result, StyleGAN generates high-quality and high-resolution images (up to 1024×1024 pixels) with detailed style-level stochastic variations. However, all images above 64×64 resolution show water droplet artifacts in the feature map, often visible in the generated output image (Karras et al, 2020b). Additionally, the progressive growing technique used in all versions of StyleGAN produces phase artifacts, where some details are stuck to the same location regardless of slight movements of the parent object, as shown in Figure 1.

D.2 StyleGAN2

StyleGAN2 (Karras et al, 2020b; NVLabs, 2021b), published in 2020, is a revised version of StyleGAN proposed to improve the image quality and remove all artifacts. First, Karras *et al.* replaced in the generator all AdaIN normalization (Huang and Belongie, 2017; Dumoulin et al, 2016, 2018; Ghiasi et al, 2017) - causing the droplet artifacts - with estimated statistics (Glorot and Bengio, 2010; He et al, 2015). Second, artifacts related to progressive growing (Karras et al, 2017) are reduced by using a modified version[17] of a hierarchical (Denton et al, 2015; Zhang et al, 2017a, 2018a) generator: Multi-Scale Gradients for GAN (MSG-GAN) (Karnewar and Wang, 2020) with skip connections (Ronneberger et al, 2015). Skip connections are used to connect matching resolutions between both networks. In parallel, residual networks (Gulrajani et al, 2017; He et al, 2016; Miyato et al, 2018) have shown benefits in the discriminator. Both alternatives replace StyleGAN's generator (synthesis network), and discriminator networks' feedforward design (Huang et al, 2020) respectively. Compared to StyleGAN, the new revision improves the training performance[18] by 40%, equivalent to 61 img/s.
However, despite the image quality improvements, tens of thousands of images with obvious variations are still required for the GAN training. Otherwise, it leads to discriminator overfitting and a training divergence (Karras et al, 2020a; Arjovsky and Bottou, 2017). Thus, acquiring this amount of varying dataset is sometimes unfeasible, as previously explained in Section 1.

Appendix E Fine-Grained Image Generation

E.0.1 InfoGAN

Information Maximization GAN (InfoGAN) (Chen et al, 2016) is a GAN that utilizes unsupervised training to disentangle common visual concepts between small subsets of the latent variables, such as the presence of objects, lighting, object azimuth, pose, elevation, etc. By doing so, InfoGAN maximizes the mutual information between the latent variables and the generated images, thereby increasing the variation in the generated dataset.

E.0.2 FineGAN

FineGAN, proposed by Singh *et al.* (Singh et al, 2019), is an architecture that disentangles the background, object shape, and object appearance hierarchically without the use of masks or fine-grained annotations. FineGAN iteratively executes in three stages: Background stage, parent stage, and child stage, where the object of interest's features, such as appearance and shape (parent stage), are combined with the previously extracted background (background stage) and then colorized with a texture (child stage) to perform FineGAN generation.
Limitations: However, FineGAN does not support conditioning on real images and only supports sampling from latent codes. Therefore, before using FineGAN, additional work to extract the background, object pose, and appearance's latent code is required to support image-conditioned generation (Li et al, 2020c).

E.0.3 MixNMatch

Li *et al.* developed MixNMatch (Li et al, 2020c; Yuheng-Li, 2020) which is built on top of FineGAN (Singh et al, 2019) and does not only allow sampled latent codes, but also real images to be used for image generation. Unlike previous fine-grained GAN architectures such as MUNIT (Huang et al, 2018), FusionGAN (Joo et al, 2018), and other disentangling techniques (Lee et al, 2018; Lorenz et al, 2019; Xiao et al, 2019), which focus on only two features: appearance and pose, MixNMatch simultaneously disentangles four factors: background, object pose, object shape, and object texture with minimal supervision. Unlike other approaches that require strong supervision annotations, such as key points, pose, masks, etc. (Peng et al, 2017; Balakrishnan et al, 2018; Ma et al, 2018b; Esser et al, 2018), MixNMatch uses only bounding box annotations to model the backgrounds, as all training images have an object of

[17] A modified version of MSG-GAN is developed to generate mipmap (Williams, 1983) instead of an image. In computer graphics, mipmaps, or pyramids, are a series of pre-computed and optimized images, each representing the previous image at progressively lower resolutions.

[18] The training is executed on NVIDIA DGX-1 with 8 Tesla V100 GPUs.

interest. Once the background generator model is trained, no bounding boxes are needed for image generation.

The MixNMatch training process consists of two stages: (1) In the first stage, the "code mode," MixNMatch takes up to 4 images and encodes them into four latent codes to generate realistic images with high accuracy. On the one hand, the shape's latent code space capacity is too small to handle unique 3D shape variations such as boxes, STRs, trolleys, etc. On the other hand, the small capacity of the shape code limits the generation of pixel-level shape and pose details, which is handled in the second stage. (2) In the second stage, the "feature mode," MixnMatch maps the image to a higher-dimensional feature space to preserve the shape and pose spatially-aligned details. Then, the FineGAN 3-stages pipeline is executed for conditional MixNMatch generation. In Figure 15, we can see the combination of the KLT box texture, ground texture, and ground color features in the generated output.

Appendix F GAN SOTA Datasets

In this section, we review the mentioned datasets in Tables 1, 2, and 3 with brief description and their downloadable links.

Appendix G Style Transfer Motivation

Style transfer (Gatys et al, 2015b; Jing et al, 2019) is one of the first and most known I2I translations to automate pastiche (Dumoulin et al, 2016). When talking about style transfer, we consider three types of images: two inputs and one output (Chen and Jia, 2021; Dumoulin et al, 2016):

1. Content image: Style transfer preserves the high-level semantic features of the content input, which are noted as invariant features, e.g., contrast, brightness, shape, etc.
2. Style image: Style transfer extracts style features from the second input image such as texture, contrast (Ulyanov et al, 2016b), color, etc.
3. Generated stylized image: Style transfer combines extracted style and content features into one single output.

Applying GANs as a backbone for I2I models is the most effective strategy (Chen and Jia, 2021). This section presents various techniques for GAN-based domain transfer and adaptive industrial applications that are essential and common to different fields. In I2I translation applications (Pang et al, 2021) e.g. semantic image synthesis (Park et al, 2019; Zhu et al, 2020; Tang et al, 2020), image segmentation (Guo et al, 2020; Li et al, 2020b), style transfer (Yi et al, 2017; Alami Mejjati et al, 2018), image inpainting (Pathak et al, 2016; Song et al, 2018; Zhao et al, 2020a), image deblurring (Zhang et al, 2020a, 2019a; Kupyn et al, 2019), image denoising (Kim et al, 2019a; Tian et al, 2020; Tran et al, 2020), image colorization (Zhang et al, 2017c; Suárez et al, 2017; He et al, 2018), super-resolution (Ledig et al, 2017; Zhang et al, 2019c; Yuan et al, 2018), domain adaptation (Cao et al, 2018; Murez et al, 2018), etc. the generative model aims to generate images that look like the target domain distribution (Chen and Jia, 2021; Pang et al, 2021). We executed arbitrary style transfer methods from (Nakano, 2018) to apply styles of different patterns, drawings, and modalities and the original image itself on an image of stillages. As a result, the style is hardly applied all over the input image: it overrides original colors and paints the whole image with a single to minimal amount of textures, as shown in Figure G5 below. Researchers focused on increasing the number of supported styles per a single network, combining arbitrary styles with interpolations (Dumoulin et al, 2016; Ghiasi et al, 2017; Nakano, 2018), conserving original images' colors and luminance beyond the style and textures (Gatys et al, 2016a), replace the *per-pixel* loss function into a perceptual loss depending on extracted high-level features (Johnson et al, 2016), maintain photo-realistic style transfer to preserve the generated image realism similarly to the content image (Li et al, 2019; Park et al, 2020b), apply instance image style transfer (Castillo et al, 2017) because of the natural image complexity which contains a variety of distinct textures (Luan et al, 2017), etc. Yet, despite its successful contribution to artistic and painter's style transfer (Pang et al, 2021), current approaches are inefficient for stylizing into industrial image modalities without any information loss. For instance, when applying a depth image as a style, the generated output consists of

Table F1 Public datasets used in GAN SOTA for Table 1

Dataset	Description
Animal FacesHQ (AFHQ)	15,000 high-quality eye-centered 512x512 PNG images equally distributed over 3 domains cat, dog and wildlife with more than 8 breeds per domain (suitable for I2I apps). An updated version AFHQv2 exists: https://github.com/clovaai/stargan-v2
BreCaHAD	162 breast cancer histopathology TIFF 1360x1024 images: https://bmcresnotes.biomedcentral.com/articles/10.1186/s13104-019-4121-7
Conceptual Captions (CC)	3M of various styles image-caption pairs harvested from the web's HTML attributes: https://github.com/google-research-datasets/conceptual-captions
Conceptual 12M (CC12m)	12M image-text pairs covering various and diverse concepts: https://github.com/google-research-datasets/conceptual-12m
CelebA	200K celebrity images covering large pose variations and background clutter with rich 40 attribute annotations: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
Chairs	1,000 different rendered 3D chair models: https://www.di.ens.fr/willow/research/seeing3Dchairs/
CIFAR-10	60,000 images at 32x32 resolution divided over 10 mutually exclusive classes of transportation vehicles and animals: https://www.cs.toronto.edu/~kriz/cifar.html
Cityscapes	Stereo video sequences recorded in street scenes from 50 cities with pixel-level annotations of 30 classes: https://www.cityscapes-dataset.com/
COCO-Stuff	164,000 selected images from MS COCO and includes 172 classes: 80 things (individual instances), 91 stuff (objects with no clear boundaries, e.g. sky, grass, street, etc.) and 1 "unlabeled": https://github.com/nightrome/cocostuff
Caltech-UCSD Birds (CUB)	11,788 annotated images of 200 birds categories, and focusing on fine-grained features, which is not familiar in most popular datasets: http://www.vision.caltech.edu/datasets/cub_200_2011/
Flickr-Faces-HQ (FFHQ)	70,000 high-quality PNG images at 1024×1024 resolution of human faces with considerable variation in terms of age, ethnicity, image background, and accessories such as eyeglasses, sunglasses, hats, etc: https://github.com/NVlabs/ffhq-dataset
ImageNet	14M hand-annotated images with an average resolution of 469x387. They are organized according to the WordNet hierarchy with more than 100,000 synsets to cover most of the concepts: https://www.image-net.org/index.php
LAION-aesthetic-6+	LAION-5B is a 5,85B CLIP-filtered image-text pairs. LAION-aesthetic-V2 has 1.2B image-text pairs with an aesthetic score higher than 4.5 based on a prediction model that imitates human rating for how much they like the image on a scale of 10: https://laion.ai/blog/laion-aesthetics/
Large-scale Scene UNderstanding (LSUN)	Large-scale scene classification, including 10 scenes (dining room, bedroom, chicken, outdoor church, etc.) and 20 object categories (cat, bus, sofa, train, etc.) with varying numbers of images up to 3M per category: https://www.yf.io/p/lsun
MetFaces	1,336 high-quality 1024x1024 PNG human faces images extracted from works of art, MetFaces-U is an unaligned version: https://github.com/NVlabs/metfaces-dataset
MNIST	60,000 grayscale images of handwritten numerical digits from 0 to 9 at a resolution of 28x28: http://yann.lecun.com/exdb/mnist/
Microsoft Common Objects in Context (MS COCO)	328,000 (200K+ annotated) images of everyday objects and humans: https://cocodataset.org/
PACS	7 categories and 4 domains split as follows: photo (1,670), art painting (2,048), cartoon (2,344) and sketch (3,929): http://sketchx.eecs.qmul.ac.uk/
Redcaps	12M image-text pairs with coarse labels collected from Reddit: https://github.com/redcaps-dataset/redcaps-downloader
Sketches	20,000 unique hand drawing (by non-expert) sketches for 250 object categories of everyday objects: https://cybertron.cg.tu-berlin.de/eitz/projects/classifysketch/
Stanford Cars	16,185 images of 196 car classes: http://ai.stanford.edu/~jkrause/cars/car_dataset.html
Stanford Dogs	20,580 annotated images from ImageNet of 120 dog breeds for the sake of fine-grained image categorization: http://vision.stanford.edu/aditya86/ImageNetDogs/main.html
Street View House Numbers (SVHN)	600,000 house numbers digit images collected from Google Street View. It provides 10 object classes corresponding to 10 digits: http://ufldl.stanford.edu/housenumbers/
YFCC100M	100 million media objects, mostly photos, including metadata such as Flickr identifier, owner name, camera, title, tags, geo, media source, taken and shared from 2004 to early 2014: http://projects.dfki.uni-kl.de/yfcc100m/

Table F2 Public datasets used in GAN SOTA for Table 2

Dataset	Description
Flower	8189 images of 102 different flower categories: https://www.robots.ox.ac.uk/~vgg/data/flowers/
Oxford Describable Textures Dataset (DTD)	5,640 textures images at a resolution of 300x300 or 640x640 for 47 categories: https://www.robots.ox.ac.uk/~vgg/data/dtd/
CMP Facades	606 building facades from all over the world including 12 annotation classes + segmentations for I2I tasks: https://cmp.felk.cvut.cz/~tylecr1/facade/
Berkeley Segmentation Dataset (BSD)	Image denoising and super-resolution dataset, which includes the BSD100 subset, a classical image dataset with 100 test images of various types, such as natural images, plants, people, food, etc, and is the testing set of the Berkeley segmentation dataset BSD300: https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

Table F3 Public datasets used in GAN SOTA for Table 3 - supervised + cross-view translations

Dataset	Description
UT Zappos50K	50,025 shoes images collected from Zappos.com divided into 4 categories (shoes, sandals, slippers, boots) with some additional instance-level labels. Paired datasets exist as well: https://vision.cs.utexas.edu/projects/finegrained/utzap50k/
edges2handbags	137K paired images of Amazon Handbags and their detected edges (by HED model + post-processing): https://efrosgans.eecs.berkeley.edu/pix2pix/datasets/
edges2shoes	50K paired images of Zappos50K datasets and their computed edges as in edges2handbags: https://efrosgans.eecs.berkeley.edu/pix2pix/datasets/
night2day	20K paired images for landscapes in the day and the night, taken from Transient Attributes dataset: https://efrosgans.eecs.berkeley.edu/pix2pix/datasets/
ADE20K	More than 25K annotated images with segmentations of various scenes (indoor and outdoor). They are selected from SUN and Places databases: https://groups.csail.mit.edu/vision/datasets/ADE20K/
NYU	1,449 densely labeled pairs of recorded video sequences from indoor scenes, with RGB, depth, and segmentations: https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html
Helen	2000 human face images with annotations of the main facial components: http://www.ifp.illinois.edu/~vuongle2/helen/
Oxford-IIIT Pet	7,349 pet images with segmentations covering 37 categories of dog and cat breeds: https://www.robots.ox.ac.uk/~vgg/data/pets/
DeepFashion	800K diverse fashion images including 300K cross-pose/cross-domain pairs: https://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html
Radboud Faces (RAFD)	Images of 67 model faces with 8 emotional expressions (anger, disgust, fear, happiness, sadness, surprise, contempt and neutral). They are taken from 5 camera angles simultaneously: https://rafd.socsci.ru.nl/RaFD2/RaFD?p=main
NTU RGB-D	56,680 video samples of 120 labeled human activities and captured in 4 different modalities as RGB videos, depth map, 3D skeletal data and infrared: https://rose1.ntu.edu.sg/dataset/actionRecognition/
Senz3D	Various hand gestures captured in RGB and segmentation with Microsoft Kinect and Leap Motion sensors: https://lttm.dei.unipd.it/downloads/gesture/
Market-1501	1,501 human identities captured by 6 different cameras: https://www.v7labs.com/open-datasets/market-1501
Dayton	76,048 image pairs for ground-to-aerial cross-view translations. It includes road and aerial views at a resolution of 354x354: https://github.com/lugiavn/gt-crossview
Crossview USA (CVUSA)	Millions of pairs of ground-level and aerial/satellite images from the United States: http://mvrl.cs.uky.edu/datasets/cvusa/
Surrounding Vehicles Awareness (SVA)	Annotated video sequences of sync frontal and bird-eye view in Grand Theft Auto V (GTAV) game: https://aimagelab.ing.unimore.it/imagelab/page.asp?IdPage=19
Ego2Top	Paired egocentric and top-view videos: https://www.crcv.ucf.edu/projects/ego2top/index.php

Table F4 Public datasets used in GAN SOTA for Table 3 - unsupervised + semi-supervised

Dataset[a]	Description
CUHK Face-Sketch Database FERET (CUFSF)	1,194 persons' face images with light variations and associated with a shape-exaggerated sketch drawn by artist: http://mmlab.ie.cuhk.edu.hk/archive/cufsf/
FaceScrub	106,863 face images of 530 celebrities in various conditions taken from the internet: http://vintage.winklerbros.net/facescrub.html
Horse2Zebra	Horse (939) and zebra (1177) images from ImageNet: https://efrosgans.eecs.berkeley.edu/cyclegan/datasets/
Apple2Orange	Apple (996) and orange images from ImageNet: https://efrosgans.eecs.berkeley.edu/cyclegan/datasets/
Summer2Winter in Yosemite	Summer (1,273) and winter (854) images in Yosemite downloaded from Flickr: https://efrosgans.eecs.berkeley.edu/cyclegan/datasets/
Paintings2Photos (Monet, Cezanne, Van Gogh, Ukiyo-e)	Monet (1074), Cezanne (584), Van Gogh (401) and Ukiyo-e (1433) paintings from Wikiart and 6,853 photographs from Flickr: https://efrosgans.eecs.berkeley.edu/cyclegan/datasets/
iPhone2DSLR Flower	1,813 iPhone and 3,316 DSLR flower images: https://efrosgans.eecs.berkeley.edu/cyclegan/datasets/
SYNTHIA	9,400 photo-realistic rendered images from an outdoor virtual city with semantic annotations for 13 classes: https://synthia-dataset.net/
Selfie2Anime	Real and anime human face images: https://github.com/taki0112/UGATIT
Places	10M images for more than 400 distinct scene categories (indoor, and outdoor): http://places2.csail.mit.edu/
Labeled Faces in the Wild (LFW)	13,233 face images of 5,749 people: http://vis-www.cs.umass.edu/lfw/
Artistic-Faces	160 artistic portraits with a wide range of artistic styles (by 16 different artists). They are annotated with 68 facial landmarks: https://faculty.runi.ac.il/arik/site/foa/artistic-faces-dataset.asp
Clothing Co-Parsing (CCP)	2,098 street fashion photos with pixel-level annotations and segmentations for 59 tags: https://github.com/bearpaw/clothing-co-parsing
Multi-Human Parsing (MHP)	25,403 images containing at least 2 persons and labeled with 58 semantic categories: https://lv-mhp.github.io/dataset
Photograph2Portrait[b]	6,452 images selected from CelebA and 1,811 paintings from Wikiart. https://github.com/HsinYingLee/DRIT
Cat2Dog	771 Birman cat and 1,264 husky and Samoyed dog images collected from Google Images: https://github.com/HsinYingLee/DRIT
Streetscape	155K highly varied and high-resolution (3000x4000) street images captured across 4 domains (sunny, night, cloudy, rain), and it includes detailed annotations for instance-level I2I: http://zhiqiangshen.com/projects/INIT/index.html
Lion2Tiger	Lion and tiger images collected from Animal With Attributes (AWA) dataset: https://cvml.ista.ac.at/AwA/
Cheetah, Cow, Lion Rhino, Bear, Wolf	10,000 rendered images (Wu et al, 2019) using SMALR 3D animal models from https://sketchfab.com/SMALR

[a]Public datasets can also be used to retrieve translation datasets such as different dog breeds from ImageNet, or face attributes from CelebA, etc., or can be used with other public datasets such as SYNTHIA or GTAV and Cityscape for outdoor scenes (Liu et al, 2017).

[b]It is characterized by a lower domain gap compared to Cat2Dog, turning the translation problem into an identity conservation issue

a sharpened grayscale image with depth information absence. Additionally, a segmentation-based style transfers the segmentation's colors into the whole content image.

Appendix H SR Briefing

In Table H5, we compared the latest different GAN-base SR technics. Recently, the latest Real-ESRGAN+ has shown superiority over SATO's SR.

Table H5 GAN-based Super Resolution Brief

GAN-based SR	Advantages	Drawbacks
SRGAN (Ledig et al, 2017)	First GAN-based SR architecture to recover 4 × down-sampled images	Compared to CNN-based architecture, it does not pursue a better quantitative measure, e.g. PSNR + Prone to distortion and many artifacts
SRFeat (Park et al, 2018)	Generates high-frequency features	Loss function layers must vary from one image to another according to its content
CinCGAN (Yuan et al, 2018)	Unsupervised Training: unpaired data can be used. No need to predefine/synthesize degradation. Recovers noisy and blurred LR images	Complex architecture, difficult and unstable training (Wang et al, 2020c)
ESRGAN (Wang et al, 2018b)	Improves SRGAN's 3 main components	Low degradation space for generating training data, thus, some artifacts remain.
Real-ESRGAN (Wang et al, 2021b)	Applied a second-order degradation model and 2D sinc filter to reduce common ringing and overshoot artifacts	Reduced overshoot artifacts may remain
Real-ESRGAN+ (Wang et al, 2021b)	Improves Real-ESRGAN's SR sharpness and reduces remaining overshoot artifacts. Supports different scales.	Inefficient for text and human face recovery.

Appendix I Additional Results

In this section, we present additional experimentation and extended results for some of the GAN architectures mentioned in the paper, such as: conditional StyleGAN3, IC-GAN, ccIC-GAN, Instance2Color, Color2Depth, image de-filtering, and image expansion in Figures I6, I7, I8, I9, I10, I11, and I12 respectively.

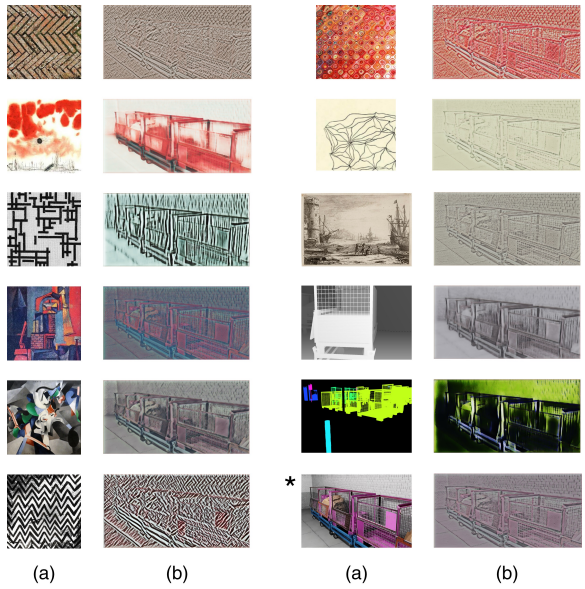


Fig. G5 Style transfer experiments based on (Nakano, 2018): (a) Style image (b) Stylized output

Fig. I6 StyleGAN3 conditional image generation for 8 logistic assets (from top to bottom): load carrier box aka. KLT Box, Trolly, Forklift, Jack, Pallet, Stillage, STR and Electrical Jack

Fig. I7 IC-GAN model trained from scratch: (a) Conditional instance (b) Generated output

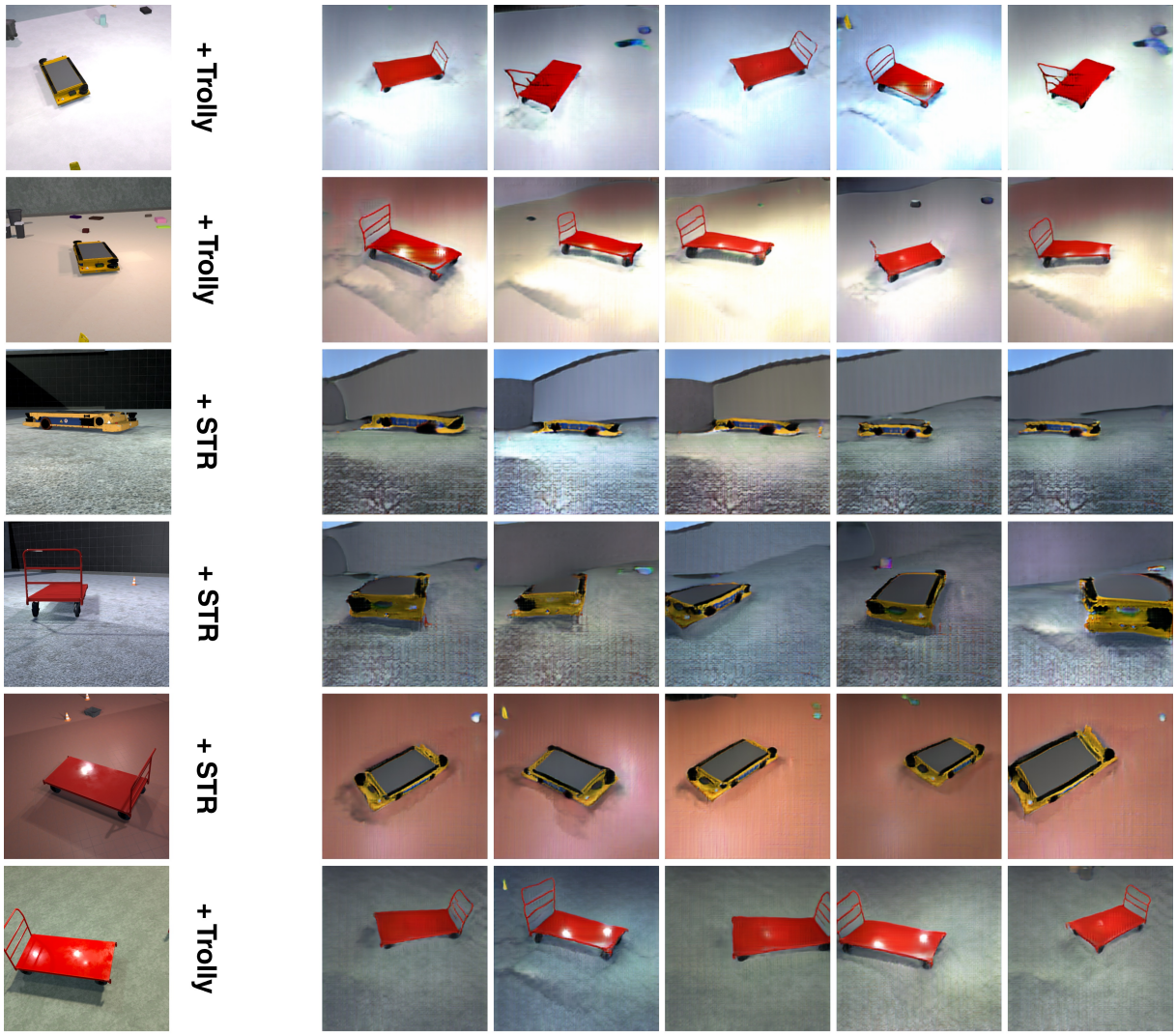


Fig. I8 ccIC-GAN model trained from scratch: (a) Conditional input (b) Output

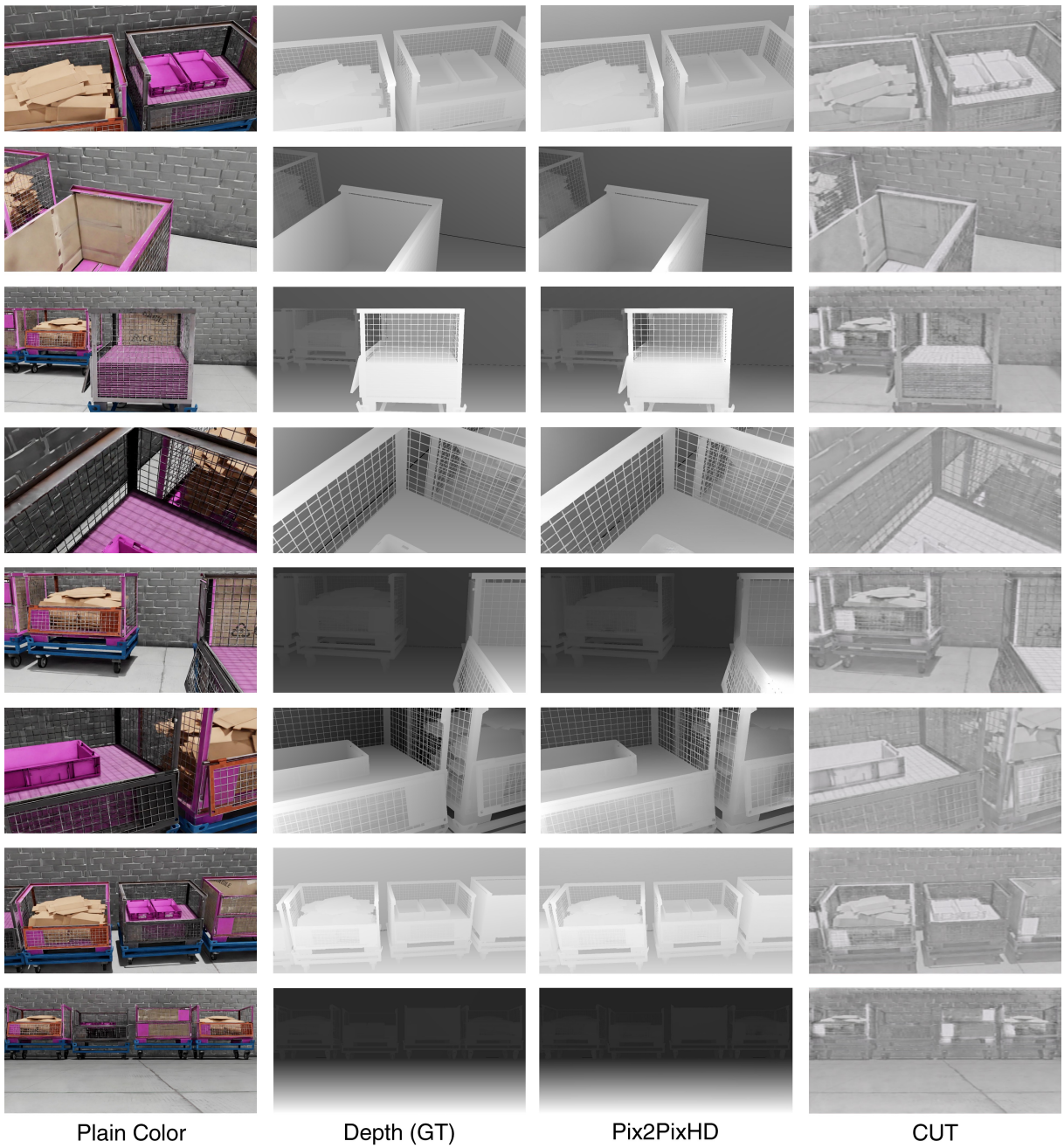


Fig. I9 Comparing domain transfer from instance segmentation image into a plain color image using Pix2PixHD and CUT (200 epochs and 1500 paired images: 3000 total images)

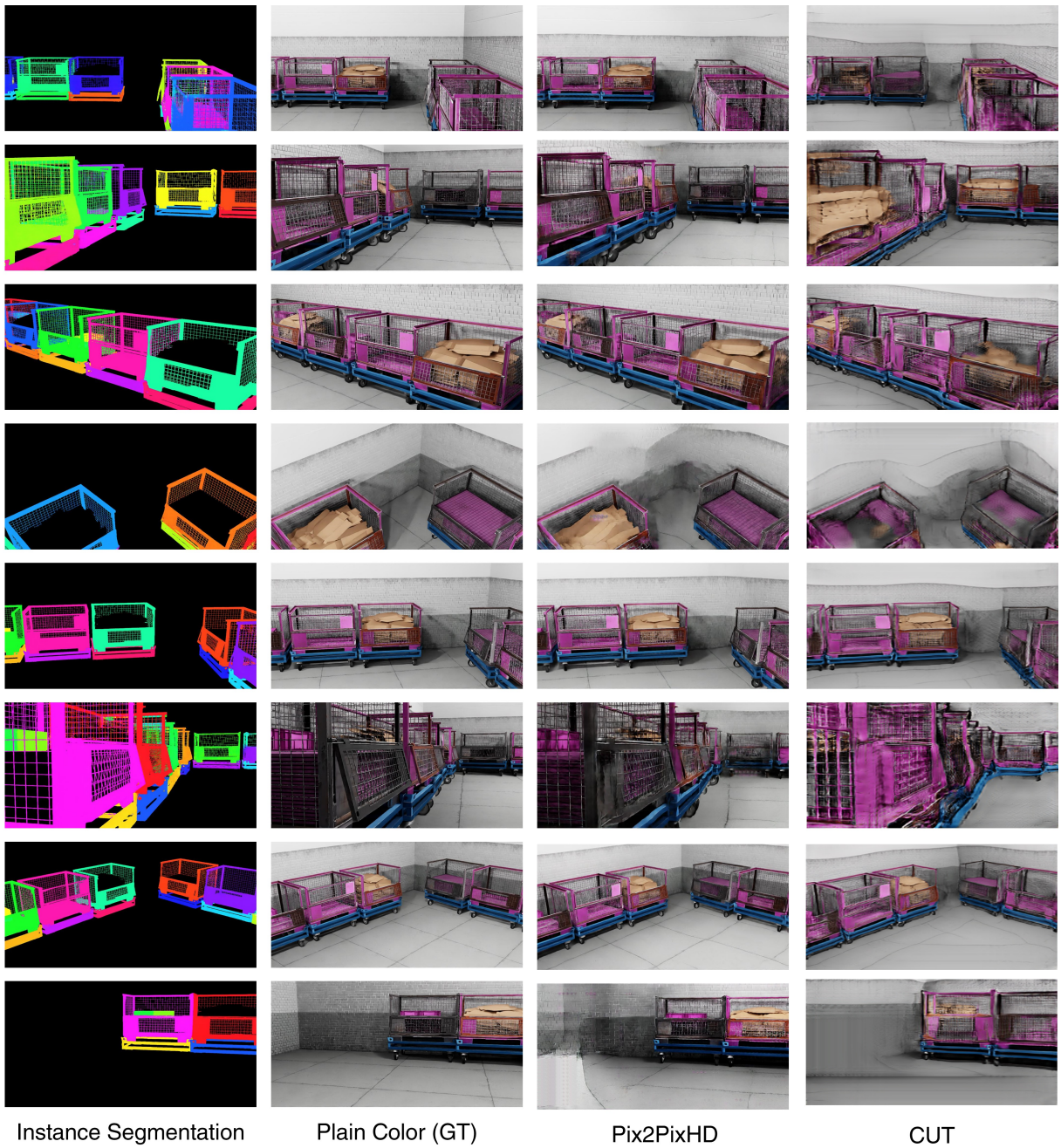


Fig. I10 Comparing domain transfer from plain color image into a depth image using Pix2PixHD and CUT (200 epochs and 1500 paired images: 3000 total images)

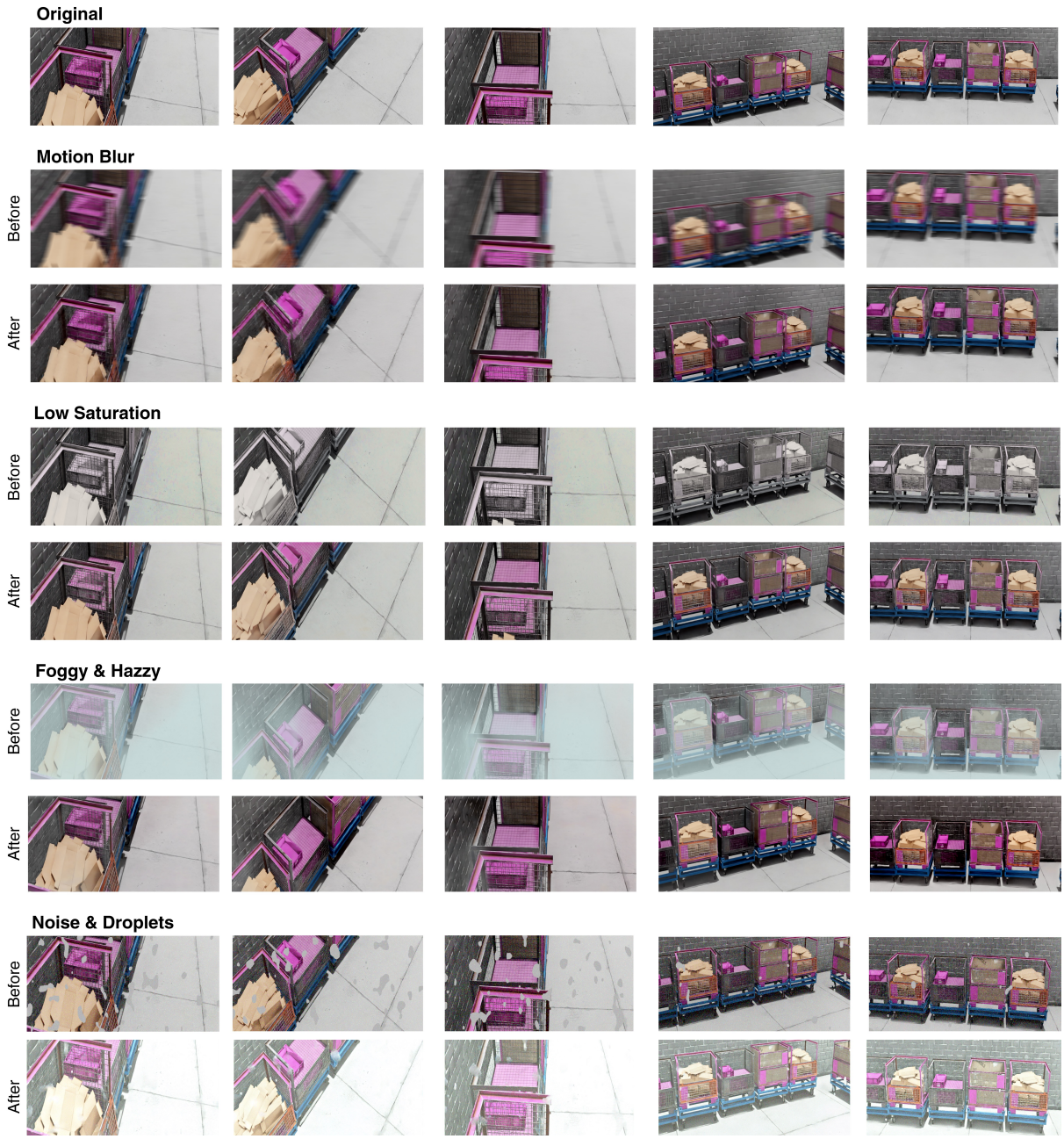


Fig. I11 Additional results for image deblurring, saturating, dehazing and denoising

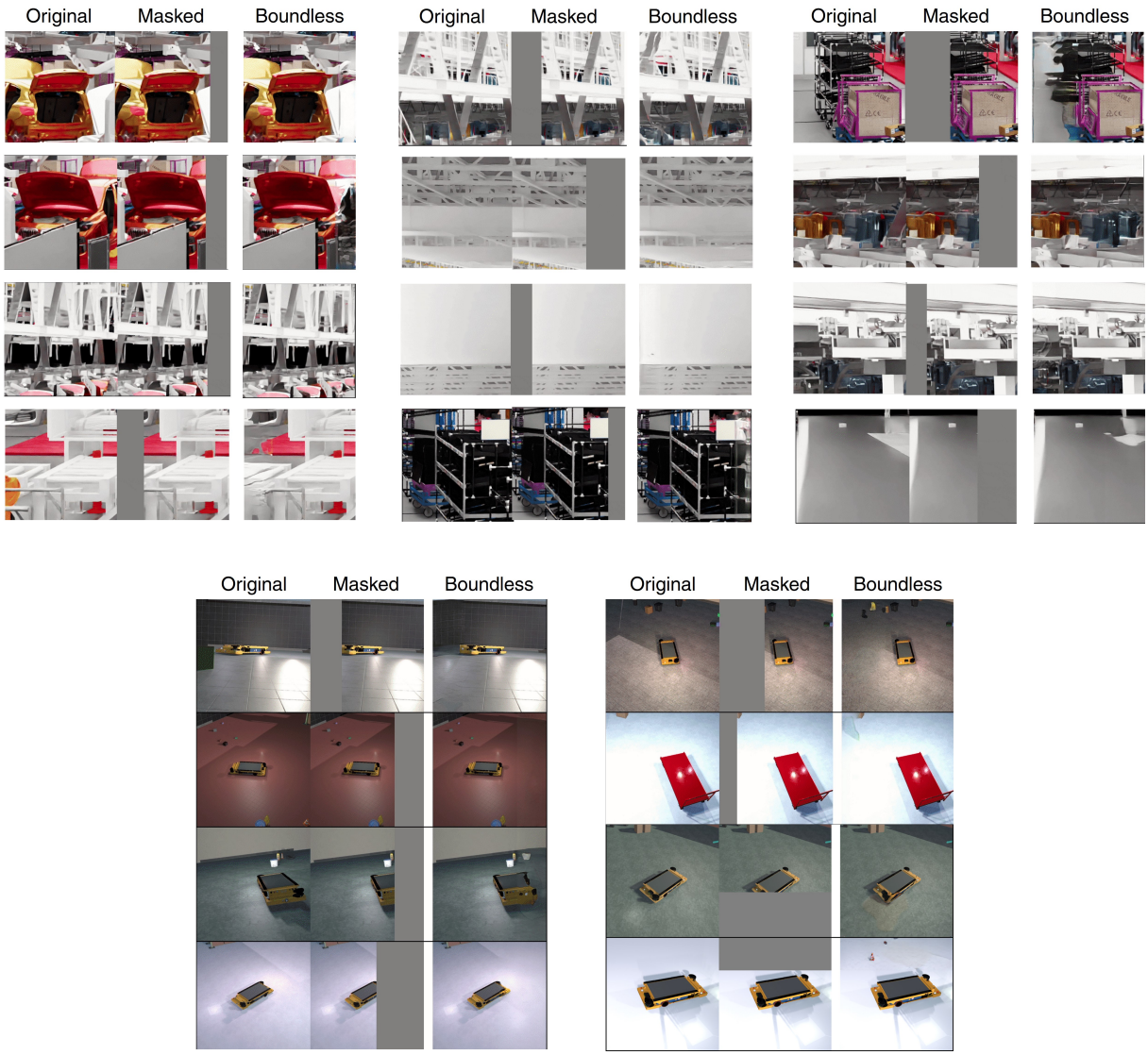


Fig. I12 Image expansion using Boundless