*algorithms*

MDPI

*Review*

# Generative Adversarial Network for Overcoming Occlusion in Images: A Survey

**Kaziwa Saleh [1,*], Sándor Szénási [2,3] and Zoltán Vámossy [2]**

[1] Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University, 1034 Budapest, Hungary
[2] John von Neumann Faculty of Informatics, Óbuda University, 1034 Budapest, Hungary; szenasi.sandor@nik.uni-obuda.hu (S.S.); vamossy.zoltan@nik.uni-obuda.hu (Z.V.)
[3] Faculty of Economics and Informatics, J. Selye University, 94501 Komárno, Slovakia
* Correspondence: kaziwa.saleh@uni-obuda.hu

**Abstract:** Although current computer vision systems are closer to the human intelligence when it comes to comprehending the visible world than previously, their performance is hindered when objects are partially occluded. Since we live in a dynamic and complex environment, we encounter more occluded objects than fully visible ones. Therefore, instilling the capability of amodal perception into those vision systems is crucial. However, overcoming occlusion is difficult and comes with its own challenges. The generative adversarial network (GAN), on the other hand, is renowned for its generative power in producing data from a random noise distribution that approaches the samples that come from real data distributions. In this survey, we outline the existing works wherein GAN is utilized in addressing the challenges of overcoming occlusion, namely amodal segmentation, amodal content completion, order recovery, and acquiring training data. We provide a summary of the type of GAN, loss function, the dataset, and the results of each work. We present an overview of the implemented GAN architectures in various applications of amodal completion. We also discuss the common objective functions that are applied in training GAN for occlusion-handling tasks. Lastly, we discuss several open issues and potential future directions.

**Keywords:** amodal completion; amodal content completion; amodal segmentation; amodal perception; order recovery; occlusion relationship; GAN; adversarial models



**Citation:** Saleh, K.; Szénási, S.; Vámossy, Z. Generative Adversarial Network for Overcoming Occlusion in Images: A Survey. *Algorithms* **2023**, *16*, 175. https://doi.org/10.3390/a16030175

Academic Editor: Frank Werner

Received: 3 February 2023
Revised: 8 March 2023
Accepted: 16 March 2023
Published: 22 March 2023



## 1. Introduction

Artificial intelligence has revolutionized the world. With the advent of deep learning and machine learning-based models, many applications and processes in our daily life have been automated. Computer vision is prominently essential in these applications, and while humans can effortlessly make sense of their surrounding, machines are far from achieving that level of comprehension. Our environment is dynamic, complex, and cluttered. Objects are usually partially occluded by other objects. However, our brain completes the partially visible objects without us being aware of it. The capability of humans to perceive incomplete objects is called amodal completion [1]. Unfortunately, this task is not as straightforward and easy for computers to achieve, because occlusion can happen in various ratios, angles, and viewpoints [2]. An object may be occluded by one or more objects, and an object may hide several other objects.

GAN is a structured probabilistic model that consists of two networks, a generator that captures the data distributions and a discriminator that decides whether the produced data come from the actual data distribution or from the generator. The two networks train in a two-player minimax game fashion until the generator can generate samples that are similar to the true samples, and the discriminator can no longer distinguish between the real and the fake samples.

Since its first introduction by Goodfellow et al. in 2014, numerous variants of GAN are proposed, mainly architecture variants and loss variants [3]. The modifications in the first category can either be in the overall network architecture such as progressive GAN (PROGAN) [4], in representation of the latent space such as conditional GAN (CGAN) [5], or in modifying the architecture toward a particular application as in CycleGAN [6]. The second category of variants encompasses modifications that are introduced to the loss functions and regularization techniques such as the Wasserstein GAN (WGAN) [7] and PatchGAN [8].

Despite the various modifications, GAN is challenging to train and evaluate. However, due to its generative power and outstanding performance, it has a significantly large number of applications in computer vision, bio-metric systems, medical field, etc. Therefore, there are a considerable number of reviews carried out on GAN and its application in different domains (shown in Section 3). There are a limited number of existing reviews that briefly mention overcoming occlusion in images with GAN. Therefore, in this survey we concentrate on the applications of GAN in amodal completion in detail. In summary, the contributions of this survey paper are:

1. We survey the literature for the available frameworks where they utilize GAN in one or more aspects of amodal completion.
2. We discuss in detail the architecture of existing works and how they have incorporated GAN in tackling the problems that occur from occlusion.
3. We summarize the loss function, the dataset, and the reported results of the available works.
4. We also provide an overview of prevalent objective functions in training the GAN model for amodal completion tasks.
5. Finally, we discuss several directions for the future research in tasks of occlusion handling wherein GAN can be utilized.

The term "occlusion handling" is polysemous in the computer vision literature. In object tracking, it mostly refers to the ability of the model to address occlusions and resume tracking the object once it re-appears in the scene [9]. In classification and detection tasks, the term indicates determining the depth order of the objects and the occlusion relationship between them [10]. Other works such as [11,12] define occlusion handling as the techniques that interpolate the blank patches in an object, i.e., content completion. However, we believe that, in order to enable a model to address occlusions, it needs the same tasks defined in amodal completion. Therefore, in this survey we use "amodal completion" and "occlusion handling" interchangeably.

As a limitation, we only focus on occlusion handling in a single 2D image. Therefore, occlusion in 3D images, stereo images, and video data are out of the scope of this work. Additionally, we emphasize on the GAN component of each architecture we reviewed. As GAN is applied for various tasks in different problems, it is difficult to carry out a systematic comparison of the existing models. Each model is evaluated on a different dataset using a different evaluation metric for a different task. In some cases, the papers do not assess the performance of GAN. In those cases, we present the result of the entire model.

The rest of this document is organized as follows: the methodology for conducting this survey is presented in Section 2. Next, Section 3 mentions the related available articles in the literature. Section 4 introduces the fundamental concepts about GAN and its training challenges, and the aspects of amodal completion. Afterward, Section 5 presents the problems in amodal completion and how GAN has been applied to address them. The common loss functions in GAN for amodal completion are discussed in Section 6. In Sections 7 and 8, future directions and key findings of this survey article are presented. Finally, conclusions are enunciated in Section 9.

## 2. Methodology

To perform a descriptive systematic literature review, we begin by forming the research questions which this survey attempts to answer. The questions are (1) what are

the challenges in amodal completion? (2) how are GAN models applied to address the problems of amodal completion? Based on the formulated questions, the search terms are identified to find and collect relevant publications. The search keywords are "GAN AND occlusion", "GAN AND amodal completion", "GAN AND occlusion handling", "GAN for occlusion handling", and "GAN for amodal completion".

We inspect several research databases, such as IEEE Xplore, Google Scholar, Web of Science, and Scopus. The list of the returned articles from the search process is sorted and refined by excluding the publications that do not satisfy the research questions. The elimination criteria are as follows: the research article addresses aspects of occlusion handling but do not employ GAN; GAN is used in applications other than amodal completion; the authors have worked on occlusion in 3D data, or video frames. Subsequently, each of the remaining publications in the list is investigated and summarized. The articles are examined for the GAN architecture, the objective function, the dataset, the results, and the purpose of using GAN.

## 3. Related Works

**Occlusion:** Handling occlusion has been studied in various domains and applications. Table 1 shows the list of published surveys and reviews of occlusion in several applications. A survey of occlusion handling in generic object detection of still images is provided in [2], focusing on challenges that arise when objects are occluded. Similarly, the most recent survey article by the authors of [13] provides the taxonomy of problems in amodal completion from single 2D images. However, none of those review articles concentrate on the applications of GAN for overcoming occlusion particularly. Other works have focused on occlusion in specific scopes, such as object tracking [14,15], pedestrians [16,17], human faces [18–22], automotive environment [23,24], and augmented reality [25]. In contrary, we review the articles that address occlusion in single 2D images.

**Table 1.** Existing survey articles about occlusion that were published between 2017 and 2022.

| # | Title | Pub. | Year |
|---|---|---|---|
| 1 | Multiple camera based multiple object tracking under occlusion: A survey [14] | IEEE | 2017 |
| 2 | Facial expression analysis under partial occlusion: A survey [18] | ACM | 2018 |
| 3 | Occlusion detection and restoration techniques for 3D face recognition: a literature review [19] | Springer | 2018 |
| 4 | Overcoming occlusion in the automotive environment—A review [23] | IEEE | 2019 |
| 5 | A comprehensive survey on multi object tracking under occlusion in aerial image sequences [15] | IEEE | 2019 |
| 6 | A Survey on Occluded Face recognition [26] | ACM | 2020 |
| 7 | Occlusion Handling in Generic Object Detection: A Review [2] | IEEE | 2021 |
| 8 | Occlusion Handling in Augmented Reality: Past, Present and Future [25] | IEEE | 2021 |
| 9 | A survey of face recognition techniques under occlusion [20] | Wiley | 2021 |
| 10 | Survey of pedestrian detection with occlusion [16] | Springer | 2021 |
| 11 | Occlusion Handling and Multi-scale Pedestrian Detection Based on Deep Learning: A Review [17] | IEEE | 2022 |
| 12 | Image Amodal Completion: A Survey [13] | arXiv | 2022 |
| 13 | A Literature Survey of Face Recognition Under Different Occlusion Conditions [21] | IEEE | 2022 |

**Generative Adversarial Network:** Due to their power, GANs are ubiquitous in computer vision research. Due to the growing body of published works in GAN, there are several recent surveys and review papers in the literature investigating its challenges, variants, and applications. Table 2 contains a list of survey articles that have been published in the last five years. The list does not include papers that specifically focus on GAN applications outside the computer vision field.

The authors in [27–32] discuss the instability problem of GAN with the various techniques and improvisations that have been designed to stabilize its training. Adversarial attack can be carried out against machine learning models by generating an input sample that leads to unexpected and undesired results by the model. Sajeeda et al. [27] investigate the various defense mechanisms to protect GAN against such attacks. Li et al. [33] summarize the different models into two groups of GAN architectures: the two-network

models and the hybrid models, which are GANs combined with an encoder, autoencoder, or variational autoencoder (VAE) to enhance the training stability. The authors of [34,35] explore the available evaluation metrics of GAN models. Other works have discussed the application of different GAN architectures for computer vision [36,37], image-to-image translation [38,39], face generation [40,41], medical field [29,42–44], person re-identification (ReID) [45], audio and video domains [29], generating and augmenting training data [46,47], image super-resolution [39,48], and other real-world applications [39,45,49,50]. Some of the mentioned review articles discuss the occlusion handling as an application of GAN very briefly, without detailing the architecture, loss functions, and the results.

**Table 2.** Available survey articles about GAN that were published between 2017 and 2022.

| # | Title | Pub. | Year |
|---|---|---|---|
| 1 | Generative adversarial networks: introduction and outlook [37] | IEEE | 2017 |
| 2 | Comparative study on generative adversarial networks [51] | arXiv | 2018 |
| 3 | Generative adversarial networks: An overview [52] | IEEE | 2018 |
| 4 | Recent progress on generative adversarial networks (GANs): A survey [34] | IEEE | 2019 |
| 5 | How generative adversarial networks and their variants work: An overview [32] | ACM | 2019 |
| 6 | Generative adversarial networks (GANs): An overview of theoretical model, evaluation metrics, and recent developments [35] | arXiv | 2020 |
| 7 | Generative adversarial network technologies and applications in computer vision [36] | Hindawi | 2020 |
| 8 | Generative adversarial networks in digital pathology: a survey on trends and future potential [42] | Elsevier | 2020 |
| 9 | Deep generative adversarial networks for image-to-image translation: A review [38] | MDPI | 2020 |
| 10 | A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications [50] | IEEE | 2021 |
| 11 | Generative adversarial network: An overview of theory and applications [49] | Elsevier | 2021 |
| 12 | The theoretical research of generative adversarial networks: an overview [33] | Elsevier | 2021 |
| 13 | Generative adversarial networks (GANs) challenges, solutions, and future directions [28] | ACM | 2021 |
| 14 | Generative adversarial networks: a survey on applications and challenges [31] | Springer | 2021 |
| 15 | A survey on generative adversarial networks for imbalance problems in computer vision tasks [46] | Springer | 2021 |
| 16 | Generative Adversarial Networks and their Application to 3D Face Generation: A Survey [41] | Elsevier | 2021 |
| 17 | Applications of generative adversarial networks (GANs): An updated review [45] | Springer | 2021 |
| 18 | Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy [3] | ACM | 2022 |
| 19 | Exploring Generative Adversarial Networks and Adversarial Training [27] | Elsevier | 2022 |
| 20 | Generative Adversarial Networks for face generation: A survey [40] | ACM | 2022 |
| 21 | Generative Adversarial Networks: A Survey on Training, Variants, and Applications [30] | Springer | 2022 |
| 22 | Augmenting data with generative adversarial networks: An overview [47] | IOS | 2022 |
| 23 | A Survey on Training Challenges in Generative Adversarial Networks for Biomedical Image Analysis [43] | arXiv | 2022 |
| 24 | Attention-based generative adversarial network in medical imaging: A narrative review [44] | Elsevier | 2022 |
| 25 | Generative adversarial networks for image super-resolution: A survey [48] | arXiv | 2022 |
| 26 | Generic image application using GANs (Generative Adversarial Networks): A Review [39] | Springer | 2022 |
| 27 | A Survey on Generative Adversarial Networks: Variants, Applications, and Training [29] | ACM | 2022 |

In this paper, we focus on the works that combine the two above-mentioned topics. Specifically, we want to present the works that have been carried out to tackle the problems that arise from occlusion using GAN. However, depending on the nature of the problems, the applicability of GAN varies. For example, in amodal appearance generation, GAN is the optimal choice of architecture. Comparably, in amodal segmentation and order recovery tasks, it is less used.

## 4. Background

### *4.1. Generative Adversarial Network*

GAN is an unsupervised generative model that contains two networks, namely a generator and a discriminator. The two networks learn in an adversary manner similar to the min–max game between two players. The generator tries to generate a fake sample that the discriminator cannot distinguish from the real sample. On the other hand, the discriminator learns to determine whether the sample is real data or generated. The generator $G$ takes a random noise $z$ as input. It learns a probability distribution $p_g$ over

data $x$ to generate fake samples that imitate the real data distribution ($p_{data}$). Then, the generated sample is forwarded to the discriminator $D$ which outputs a single scalar that labels the data as real or fake (Figure 1). The classification result is used in training $G$ as gradients of the loss. The loss guides $G$ to generate samples that are less likely and more challenging to be labeled as fake by the $D$. Overtime, $G$ becomes better in generating more realistic samples that would confuse $D$, and $D$ becomes better at detecting fake samples. They both try to optimize their objective functions, in other words, $G$ tries to minimize its cost value and $D$ tries to maximize its cost value.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z)))] \quad (1)$$

Equation (1) was designed by Goodfellow et al. [53] to compute the cost value of GAN where $x$ is the real sample from the training dataset, $G(z)$ is the generated sample, and $D(x)$ and $D(G(z))$ are the discriminator's verdict that $x$ is real and the fake sample $G(z)$ is real.

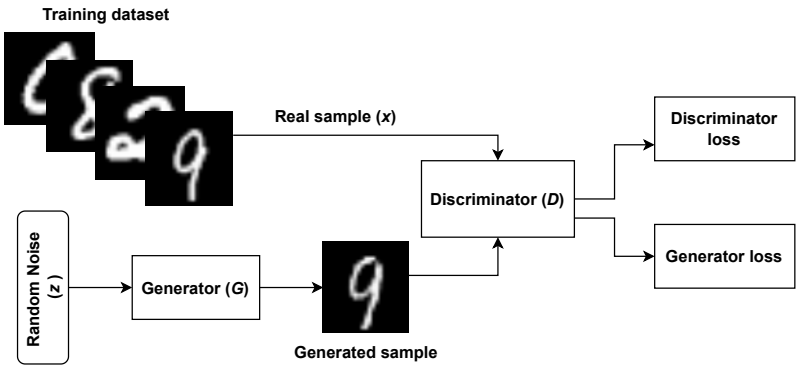


**Figure 1.** Architecture of the original GAN [53].

There are numerous variations of the original GAN. Among the most prominent ones are CGAN, WGAN, and Self-Attention GAN (SAGAN) [54]. CGAN extends the original GAN by taking an additional input which is usually a class label. The label conditions the generated data to be of a specific class. Therefore, the loss function in (1) becomes as follows:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x \mid c)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z \mid c)))] \quad (2)$$

where $c$ is the conditional class label.

In order to prevent the vanishing gradient and mode collapse problems (discussed below), WGAN applies an objective function that implements the Earth-Mover (EM) [55] distance for comparing the generated and real data distributions. EM helps in stabilizing GAN's training and the equilibrium between the generator and the discriminator. If the gradient of the loss function becomes too large, WGAN will employ weight clipping. WGAN Gradient Penalty (WGAN-GP) [56] extends WGAN by introducing a penalty term instead of the weight clipping to enhance the training stability, convergence power, and output quality of the network. Moreover, SAGAN applies an attention mechanism to extract features from a broader feature space and capture global dependencies instead of the local neighborhoods. Thus, SAGAN can produce high-resolution details in data as it borrows cues from all feature locations in contrast to the original GAN that depends on only spatially local points.

In theory, both $G$ and $D$ are expected to converge at the Nash equilibrium point. However, in practice this is not as simple as it sounds. Training GANs is challenging, because they are unstable and difficult to evaluate. GANs are notorious for several issues, which are already covered intensively in the literature; therefore, we will only discuss them briefly below.

#### 4.1.1. Achieving Nash Equilibrium

In game theory, Nash equilibrium is when none of the players will change their strategy no matter what the opponents do. In GAN, the game objective changes as the networks take turn during the training process. Therefore, it is particularly difficult to obtain the desired equilibrium point due to the adversarial behavior of its networks. Typically, gradient descent is used to find the minimum value of the cost function during training. However, in GAN, decreasing the cost of one network leads to the increase in the cost of the other network. For instance, if one player minimizes $xy$ with regard to $x$ and another player minimizes $-xy$ with regard to $y$, gradient descent reaches a stable sphere, but it does not converge to the equilibrium point which is $x = y = 0$ [57].

#### 4.1.2. Mode Collapse

One of the major problems with GANs is that they are unable to generalize well. This poor generalization leads to mode collapse. The generator collapses when it cannot generate large diverse samples known as complete collapse, or it will only produce a specific type (or subset) of target data that will not be rejected by the discriminator as being fake, known as partial collapse [53,57].

#### 4.1.3. Vanishing Gradient

GAN is challenging to train due to the vanishing gradient issue. The generator stops learning when the gradients of the weights of the initial layers become extremely small. Thus, the discriminator confidently rejects the samples produced by the generator [58].

#### 4.1.4. Lack of Evaluation Metrics

Despite the growing progress in the GAN architecture and training, evaluating it remains a challenging task. Although several metrics and methods have been proposed, there is no standard measure for evaluating the models. Most of the available works propose a new technique to assess the strength and the limitation of their model. Therefore, finding a consensus evaluation metric remains an open research question [59].

### *4.2. Amodal Completion*

Amodal completion is the natural ability of humans to discern the physical objects in the environment even if they are occluded. Our environment contains more partially visible or temporarily occluded objects than fully visible ones. Hence, the input to our visual system is mostly incomplete and segmented. Yet, we innately and effortlessly imagine the invisible parts of the object in our mind and perceive the object as complete [1]. For instance, if we only see a half of stripped legs in the zoo, we can tell that there is a zebra in that territory.

As natural and seamless this task is for humans, for computers it is challenging yet essential. This is because the performance of most computer vision-related real-world applications drop when objects are occluded. For example, in autonomous driving, the vehicle must be able to recognize and identify the complete contour of the objects in the scene to avoid accidents and drive safely.

Our environment is complex, cluttered, and dynamic. An object may be behind one or more other objects, or an object may hide one or more other objects. Thus, possible occlusion patterns between objects are endless. Therefore, the shape and appearance of occluded objects are unbounded.

Whenever a visual system requires de-occlusion, there are three sub-tasks involved in the process (Figure 2). Firstly, inferring the complete segmentation mask of the partially visible objects, including the hidden region. Secondly, predicting and reconstructing the RGB content of the occluded area based on the visible parts of the object and/or the image. Often, these two sub-tasks require the result of the third sub-task, which determines the depth order of the objects and the relationship between them, i.e., which object is

the occluder and which one is the occludee. Several of the existing works address these sub-tasks simultaneously.

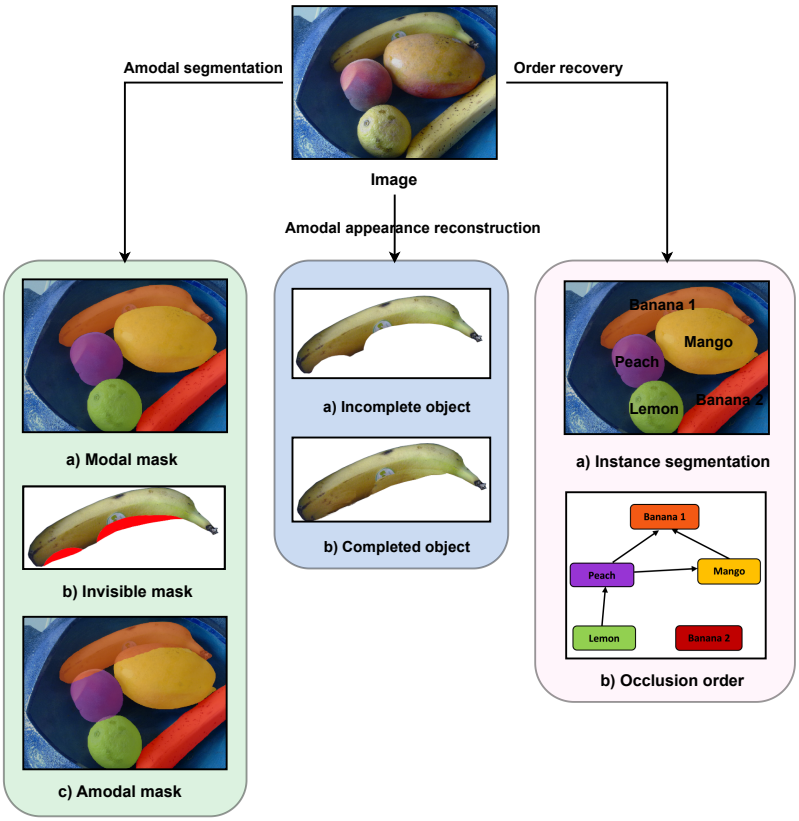


**Figure 2.** The three sub-tasks in amodal completion.

Designing and training a model that could perform any/all of the above-mentioned sub-processes presents several challenges. In the following section, we explore the existing works in the literature wherein a GAN architecture is implemented to address those obstacles.

## 5. GAN in Amodal Completion

The taxonomy of the challenges in amodal completion is presented by Ao et al. [13]. In the following sections, we present how GAN has been used to address each challenge. In exploring the existing research papers, we emphasized the aspects of amodal completion wherein GAN was utilized, not the original aim of the paper.

### *5.1. Amodal Segmentation*

Image segmentation tasks such as semantic segmentation, instance segmentation, or panoptic segmentation solely predict the visible shape of the objects in a scene. Therefore, these tasks mainly operate with modal perception. Amodal segmentation, on the other hand, works with amodal perception. It estimates the shape of an object beyond the visible region, i.e., the visible mask (also called the modal mask) and the mask for the occluded region, from the local and the global visible visual cues (see Figure 3).

Amodal segmentation is rather challenging, especially if the occluder is of a different category (e.g., the occlusion between vehicles and pedestrians). The visible region may not hold sufficient information to help in determining the whole extent of the object. Contrariwise, if the occluder is an instance of the same category (e.g., occlusion between pedestrians), since the features of both objects are similar, it becomes difficult for the model to estimate where the boundary of one object ends and the second one begins. In either case, the visible region plays a significant role in guiding the amodal mask generation process. Therefore, most existing methods require the modal mask as input. To

alleviate the need for a manually annotated modal mask, many works apply a pre-trained instance segmentation network to obtain the visible mask and utilize it as input.

(**a**) Image. (**b**) Semantic segmentation. (**c**) Instance segmentation.

(**d**) Panoptic segmentation. (**e**) Amodal segmentation.

**Figure 3.** Different types of image segmentation.

In the following, we describe the architecture of the GAN-based models that are used in generating the amodal mask of the occluded objects.

**A two hourglass generator:** Zhou et al. [60] apply a pre-trained instance segmentation network on the input image to obtain an initial mask and feeds it to a two-stage pipeline for human deocclusion. Given the initial mask, the generator implements two hourglass modules to refine and complete the modal mask to produce the amodal mask at the end. A discriminator enhances the quality of the output amodal mask. An additional parsing result accompanies the result of the generator, which is employed by a Parsing Guided Attention (PGA) module to reinforce the semantic features of body parts at multiple scales as a part of a parsing guided content recovery network. The latter uses a combination of UNet [61] and partial convolutions [62] in generating the content of the invisible area. The additional parsing branches add extra semantic guidance, which improves the final invisible mask.

**A coarse-to-fine architecture with contextual attention:** Xiong et al. [63] firstly employ a contour detection module to extract the visible contour of an object and then complete it through a contour completion network. The contour detection module uses DeepCut [64] to segment prominence objects, and performs noise removal and edge detection to extract the incomplete contour of the object from the segmentation map. Then, the contour completion network learns to conjecture the foreground contour. The contour completion network is composed of a generator and a discriminator. The generator has a coarse-to-fine architecture, each with a similar encoder–decoder structure, except that the refinement network employs a contextual attention layer [65]. Finally, the completed contour along with the ground-truth image are fed to the discriminator which produces a score map to indicate the originality of each region in the generated contour mask and can decide whether the mask aligns with the contour of the image. The discriminator is a fully convolutional PatchGAN [8] trained with a hinge loss. The results show that the contour completion step assists in the explicit modeling of the background and the foreground layer borders, which leads to less evident artifacts in the completed foreground objects.

**A generator with priori knowledge:** The authors of [66] also utilize a pre-trained instance segmentation model to obtain the visible human mask, which is fed with the input image into a GAN-based model to produce the amodal mask of occluded humans. The model predicts the mask of the invisible region through an hourglass network structure.

The local fine features and the higher-level semantic details are aggregated in the encoding stage, and they are added to each layer's feature maps in the decoding stage. The predicted amodal mask is evaluated by a Patch-GAN discriminator. To improve the amodal segmentation outcome, some typical human poses are concatenated with the feature maps as a priori information to be used in the decoding stage. Although the a priori knowledge enhances the predicted amodal masks, it restricts the application of the model to humans with specific poses.

**A coarse-to-fine architecture with multiple discriminators:** In the applications such as visual surveillance and autonomous driving, path prediction, and intelligent traffic control, detecting vehicles and pedestrians is essential. However, these are often obstructed by other objects which makes the task of learning the visual representation of intended objects more challenging. The model in [67] aims to recover the amodal mask of a vehicle and the appearance of its hidden regions iteratively. To tackle both tasks, the model is composed of two parts: a segmentation completion module and an appearance recovery module. The first network, follows an initial-to-refined framework. Firstly, an initial segmentation mask is generated by taking an input image with occluded vehicles through a pre-trained segmentation network. Then, the input image is fed again into the next stage after it is concatenated with the output from the initial stage. The second part, in contrary to a standard GAN, has a generator with an encoder–decoder structure, an object discriminator, and an instance discriminator. To assist the model in producing more realistic masks, an additional 3D model pool is employed. This provides silhouette masks as adversarial samples which motivates the model to learn the defining characteristics of actual vehicle masks. The object discriminator, which uses a Stack-GAN structure [68], enforces the output mask to be similar to a real vehicle, whereas the instance discriminator with a standard GAN structure aims at producing an output mask similar to the ground-truth mask. The recovered mask is fed to the appearance recovery module to regenerate the whole foreground vehicle. Both modules are trained with reconstruction loss (i.e., $\mathcal{L}1$ loss) and perceptual loss. Although using the 3D model pool and multiple discriminators produces better amodal masks, when the model is tested on synthetic images with different types of synthetic occlusions, it requires multiple iterations to progressively eliminate the occlusions. However, on real images with less severe occlusions, the model is unable to refine the results beyond three iterations and its performance declines.

### *5.2. Order Recovery*

In order to apply any de-occlusion or completion process, it is essential to determine the occlusion relationship and identify the depth order between the overlapping components of a scene. Other processes such as amodal segmentation and content completion depend on the predicted occlusion order to accomplish their tasks. Therefore, vision systems need to distinguish the occluders from the occludees, and to determine whether an occlusion exists between the objects. Order recovery is vital in many applications, such as semantic scene understanding, autonomous driving, and surveillance systems.

The following works attempt to retrieve the depth order/layer order between the objects in a scene through utilizing a GAN-based architecture.

**A generator with multiple discriminators:** Dhamo et al. [69] present a method to achieve layered depth prediction and view synthesis. Given a single RGB image as input, the model learns to synthesize a RGB-D view from it and hallucinates the missing regions that were initially occluded. Firstly, the framework uses a fully-convolutional network to obtain a depth map and a segmentation mask for foreground and background elements from the input image. Depending on the predicted masks, the foreground objects are erased from the input image and the obtained depth map (RGB-D). Then, a Patch-GAN [8]-based network is used to refill the holes in the RGB-D background image that were created from removing the foreground objects. The network has a pair of discriminators to enforce inter-domain consistency. This method has data limitations, as it is difficult to obtain ground-truth layered depth images in real-world data.

Inferring the scene layout beyond the visible view and hallucinating the invisible parts of the scene is called amodal scene layout. MonoLayout, proposed in [70], provides the amodal scene layout in the form of bird's eye view (BEV) in real time. With a single input image of a road scene, the framework delivers a BEV of static (such as sidewalks and street areas) and dynamic (vehicles) objects in the scene, including the partially visible components. The model contains a context encoder, two decoders, and two discriminators. Given the input image, the encoder captures the multi-scale context representations of both static and dynamic elements. Then, the context features are shared with two decoders, an amodal static scene decoder and a dynamic scene decoder, to predict the static and dynamic objects in BEV. The decoders are regularized by two corresponding discriminators to encourage the predictions to be similar to the ground-truth representations. The context sharing within the decoders achieves better performance of amodal scene layout. MonoLayout can infer 19.6 M parameters in 32 fps. However, it needs generalization for unseen scenarios.

**A single generator and discriminator:** Zheng et al. [71] tackle the amodal scene understanding by creating a layer-by-layer pipeline (Completed Scene Decomposition Network (CSDNet)) to extract and complete RGB appearance of objects from a scene, and make sense of their occlusion relation. In each layer, CSDNet only separates the foreground elements that are without occlusion. This way, the system identifies and fills the invisible portion of each object. Then, the completed image is fed again to the model to segment the fully visible objects. In this iterative manner, the depth order of the scene is obtained, which can be used to recompose a new scene. The model is composed of a decomposition network and a completion network. The decomposition network follows Mask-RCNN [72] with an additional layer classification branch to estimate the instance masks, and determine whether an object is fully or partially visible. The predicted masks are forwarded to the completion network, which uses an encoder–decoder to complete the resultant holes in the masked image. By masking the fully visible objects in each step and the iterative completion of the objects in the scene, the earlier completion information is propagated to the later steps. Nonetheless, the model is trained on a rendered dataset; therefore, it cannot generalize well to real scenes that are unlike the rendered ones. In addition, the completion errors over the layers are accumulated, which leads to a drop in accuracy when the occlusion layers are too numerous.

On the other hand, Dhamo et al. [73] present an object-oriented model with three parts: object completion, layout prediction, and image re-composition, while the object completion unit attempts to fill the occluded area in the input RGBA image through an auto-encoder, the layout prediction uses a GAN architecture to estimate the RGBA-D (the RGBA and depth images) background, i.e., the object-free representation of the scene. The model infers the layered representation of a scene from a single image and produces a flexible number of output layers based on the complexity of the scene. However, the global and the local contexts, and the spatial relationship between the objects in the scene, are not considered.

### *5.3. Amodal Appearance Reconstruction*

Recently, there has been a significant progress in image inpainting methods, such as the works in [65,74]. However, these models recover the plausible content of a missing area with no knowledge about which object is involved in that part. On the contrary, amodal appearance reconstruction (also known as amodal content completion) models require identifying individual elements in the scene, and recognizing the partially visible objects along with their occluded areas, to predict the content for the invisible regions.

Therefore, the majority of the existing frameworks follow a multi-stage process to address the problem of amodal segmentation and amodal content completion as one problem. Therefore, they depend on the segmentator to infer the binary segmentation mask for the occluded and non-occluded parts of the object. The mask is then forwarded as input to the amodal completion module, which tries to fill in the RGB content for the missing region indicated by the mask.

Among the three sub-tasks of amodal completion, GAN is most widely used in amodal content completion. In this section, we present the usage of GAN in amodal content completion for a variety of computer vision applications.

#### 5.3.1. Generic Object Completion

GANs are unable to estimate and learn the structure in the image implicitly with no additional information about the structures or annotations regarding the foreground and background objects during training. Therefore, Xiong et al. [63] propose a model that is made up of a contour detection module, a contour completion module, and an image completion module. The first two modules learn to detect and complete the foreground contour. Then, the image completion module is guided by the completed contour to determine the position of the foreground and the background pixels. The incomplete input image, the completed contour, and the hole mask are fed to the image completion network to fill the missing part of the object. The network has a similar coarse-to-fine architecture as the contour completion module. However, the depth of the network weakens the effect of the completed contour. Therefore, the complete contour is passed to both the coarse network and the refinement network. The discriminator of the image completion network is a PatchGAN that is trained with hinge loss and requires the generated fake image or the ground-truth image with the hole mask. The experiments show that, under the guide of the contour completion, the model can generate completed images with less artifacts and complete objects with more natural boundaries. However, the model will fail to produce results without artifacts and color discrepancy around the holes due to implementing vanilla convolutions in extracting the features.

Therefore, Zhan et al. [75] use CGAN and partial convolution [62] to regenerate the content of the missing region. The authors apply the concept of partial completion to de-occlude the objects in an image. In the case of an object hidden by multiple other objects, the partial completion is performed by considering one object at a time. The model partially completes both the mask and the appearance of the object in question through two networks, namely Partial Completion Network-mask (PCNet-M) and Partial Completion Network-content (PCNet-C), respectively. A self-supervised approach is implemented to produce labeled occluded data to train the networks, i.e., a masked region is obtained by positioning a randomly selected occluder from the dataset on top of the concerned object. Then, the masked occludee is passed to the PCNet-M to reproduce the mask of the invisible area, which in turn is given to the PCNet-C. Although the self-supervised and partial completion techniques alleviate the need for annotated training data, the generated content contains the remaining of the occluder and its quality is not good if it has texture.

Ehsani et al. [76] trained a GAN-based model dubbed SeGAN. The model consists of a segmentator which is a modified ResNet-18 [77], and a painter which is a CGAN. The segmentator produces the full segmentation mask (amodal mask) of the objects including the occluded parts. On the other hand, the painter, which consists of a generator and a discriminator, takes in the output from the segmentator and reproduces the appearance of the hidden parts of the object based on the amodal mask. The final output from the generator is a de-occluded RGB image which is then fed into the discriminator. As a drawback, the model is trained on a synthetic dataset, which presents an inevitable domain gap between the training images and the real-world testing images.

Furthermore, Kahatapitiya et al. [78] aim to detect and remove the unrelated occluders, and inpaint the missing pixels to produce an occlusion-free image. The unrelated objects are identified based on the context of the image and a language model. Through a background segmentator and the foreground segmentator, the background and foreground objects are extracted, respectively. The foreground extractor produces pixel-wise annotations for the objects (i.e., thing class) and the background segmentator outputs the background objects (i.e., stuff class). Then, the relation predictor uses the annotations to estimate the relation of each foreground object to the image context based on a vector embedding of class labels trained with a language model. The result of the relation prediction can detect any

unrelated objects which are considered as unwanted occlusion. Consequently, the relations and pixel annotations of the thing class are fed into the image inpainter to mask and recreate the pixels of the hidden object. The image inpainter is based on the contextual attention model by Yu et al. [65], which employs a coarse-to-fine model. In the first stage, the mask is coarsely filled in. Then, the second stage utilizes a local and a global WGAN-GP [56] to enhance the quality of the generated output from the coarse stage. A contextual attention layer is implemented to attend to similar feature patches from distant pixels. The local and global WGAN-GP enforce global and local consistency of the inpainted pixels [65]. The contextual information helps in generating a de-occluded image; however, the required class labels of the foreground and background objects limit the applicability of the method.

#### 5.3.2. Face Completion

Occlusion is usually present in faces. The occluding objects can be glasses, scarf, food, cup, microphone, etc. The performance of biometric and surveillance systems can degrade when faces are obstructed or covered by other objects, which raises a security concern. However, compared to background completion, facial images are more challenging to complete since they contain more appearance variations, especially around the eyes and the mouth. In the following, we categorize the available works for face completion based on their architecture.

**A single generator and discriminator:** Cai et al. [79] present an Occlusion-Aware GAN (OA-GAN), with a single generator and discriminator, that alleviates the need for an occlusion mask as an input. Through using paired images with known mask of artificial occlusions and natural images without occlusion masks, the model learns in a semi-supervised way. The generator has an occlusion-aware network and a face completion network. The first network estimates the mask for the area where the occlusion is present, which is fed into the second network. The latter then completes the missing region based on the mask. The discriminator employs an adversarial loss, and an attribute preserving loss to ensure that the generated facial image has similar attributes to the input image.

Likewise, Chen et al. [80] depend on their proposed OA-GAN to automatically identify the occluded region and inpaint it. They train a DCGAN on occlusion-free facial images, and use it to detect the corrupted regions. During the inpainting process, a binary matrix is maintained, which indicates the presence of occlusion in each pixel. The detection of occluded region alleviates the need for any prior knowledge of the location and type of the occlusion masks. However, incorrect occlusion detection leads to partially inpainted images.

Facial Structure Guided GAN (FSG-GAN) [81] is a two-stage model with a single generator and discriminator. In the first part, a variational auto-encoder estimates the facial structure which is combined with the occluded image and fed into the generator of the second stage. The generator (UNet), guided by the facial structure knowledge, synthesizes the deoccluded image. A multi-receptive fields discriminator encourages a more natural and less ambiguous appearance of the output image. Nevertheless, the model cannot remove occlusion in a face image with large posture well, and it cannot correctly predict the facial structure under severe occlusions, which leads to unpleasant results.

**Multiple discriminators:** Several of the existing works employ multiple discriminators to ensure that the completed facial image is semantically valid and consistent with the context of the image. Li et al. [82] train a model with a generator, a local discriminator, a global discriminator, and a parsing network to generate an occlusion-free facial image. The original image is masked with a randomly positioned noisy square and fed into the generator which is designed as an auto-encoder to fill the missing pixels. The discriminators, which are binary classifiers, enhance the semantic quality of the reconstructed pixels. Meanwhile, the parsing network enforces the harmony of the generated part and the present content. The model can handle various masks of different positions, sizes, and shapes. However, the limitations of the model include the facts that (1) it cannot recognize the position/orientation of the face and its corresponding elements which leads

to unpleasant generative content; (2) it fails to correctly recover the color of the lips; (3) it does not capture the full spatial correlations within neighboring pixels.

Similarly, Mathai et al. [83] use an encoder–decoder for the generator, a Patch-GAN-based local discriminator, and a WGAN-GP [56]-based global discriminator to address occlusions on distinctive areas of a face and inpaint them. Consequently, the model's ability in recognizing faces improves. To minimize the effect of the masked area on the extracted features, two convolutional gating mechanisms are experimented: hard gating mechanism known as partial convolutions [62] and a soft gating method based on sigmoid function.

Liu et al. [84] also follow the same approach by implementing a generator (autoencoder), a local discriminator, and a global discriminator. A self-attention mechanism is applied in the global discriminator to enforce complex geometric constrains on the global image structure, and model long-range dependencies. The authors report the results for the facial landmark detection only, without providing the experimental data.

Moreover, Cai et al. [85] present FCSR-GAN to create a high-resolution deoccluded image from a low-resolution facial image with partial occlusions. At first, the model is pre-trained for face completion to recover the missing region. Afterward, the entire framework is trained end-to-end. The generator comprises a face completion unit and a face super-resolution unit. The low-resolution occluded input image is fed into the face completion module to fill the missing region. The face completion unit follows an encoder–decoder layout and the overall architecture is similar to the generative face completion by Li et al. [82]. Then, the occlusion-free image is fed into the face super-resolution module which adopts a SRGAN [86]. The network is trained with a local loss, a global loss, and a perceptual loss to ensure that the generated content is consistent with the local details and holistic contextual information. An additional face parsing loss and perceptual loss are computed to produce more realistic face images.

Furthermore, face completion can improve the resistance of face identification and recognition models to occlusion. The authors in [87] propose a two-unit de-occlusion distillation pipeline. In the de-occlusion unit, a GAN is implemented to recover the appearance of pixels covered by the mask. Similar to the previously mentioned works, the output of the generator is evaluated by local and global discriminators. In the distillation unit, a pre-trained face recognition model is employed as a teacher, and its knowledge is used to train the student model to identify masked faces by learning representations for recovered faces with similar clustering behaviors as the original ones. This teaches the student model how to fill in the information gap in appearance space and in identity space. The model is trained with a single occlusion mask at a time; however, in real-world instances, multiple masks cover large discriminative regions of the face.

**Multiple generators:** In contrast to the OA-GAN presented by Cai et al. [79], the authors of [88] propose a two-stage OA-GAN framework with two generators and two discriminators. While the generators ($G_1$, and $G_2$) are made up of a UNet encoder–decoder architecture, PatchGAN is adopted in the discriminators. $G_1$ takes an occluded input image and disentangles the mask of the image to produce a synthesized occlusion. $G_2$ then takes the output from $G_1$ in order to remove the occlusions and generate a deoccluded image. Therefore, the occlusion generator (i.e., $G_1$) plays a fundamental role in the deocclusion process. The failure in the occlusion generator produces incorrect images.

**Multiple generators and discriminators:** While using multiple discriminators ensures the consistency and the validity of the produced image, some available works employ multiple generators, especially when tackling multiple problems. For example, Jabbar et al. [89] present a framework known as Automatic Mask Generation Network for Face Deocclusion using Stacked GAN (AFD-StackGAN) that is composed of two stages to automatically extract the mask of the occluded area and recover its content. The first stage employs an encoder–decoder in its generator to generate the binary segmentation mask for the invisible region. The produced mask is further refined with erosion and dilation morphological techniques. The second stage eliminates the mask object and regenerates the corrupted pixels through two pair of generators and discriminators. The occluded

input image and the extracted occlusion mask are fed into the first generator to produce a completed image. The initial output from the first generator is enhanced by rectifying any missing or incorrect content in it. Two PatchGAN discriminators are implemented against the result of the generators to ensure that the restored face's appearance and structural consistency are retained. AFD-StackGAN can remove various types of occlusion masks in the facial images that cover a large area of the face. However, it is trained with synthetic data, and the incompatibility of the training images and the real-world testing images is likely.

In the same way, Li et al. [90] employ two generators and three domain-specific discriminators in their proposed framework called disentangling and fusing GAN (DF-GAN). They treat face completion as disentangling and fusing of clean faces and occlusions. This way, they remove the need for paired samples of occluded images and their congruent clean images. The framework works with three domains that correspond to the distribution of occluded faces, clean faces, and structured occlusions. In the disentangling module, an occluded facial image is fed into an encoder which encodes it to the disentangled representations. Thereafter, two decoders produce the corresponding deoccluded image and occlusion, respectively. In other words, the disentangling network learns how to separate the structured occlusions and the occlusion-free images. The fusing network, on the other hand, combines the latent representations of clean faces and occlusions, and creates the corresponding occluded facial image, i.e., it learns how to generate images with structured occlusions. However, real-world occlusions are of arbitrary shape and size, not necessarily structured.

**Coarse-to-fine architecture:** Conversely to the previously mentioned works where one output is generated, Jabbar et al. [91] propose a two-stage Face De-occlusion using Stacked Generative Adversarial Network (FD-StackGAN) model that follows the coarse-to-fine approach. The model attempts to remove the occlusion mask and fill in the affected area. In the first stage, the network produces an initial deoccluded facial image. The second stage refines the initial generated image to create a more visually plausible image that is similar to the real image. Similar to AF-StackGAN, FD-StackGAN can handle various regions in the facial images with different structures and surrounding backgrounds. However, the model is trained on a synthetic dataset but it is not tested on images with natural occlusions.

Likewise, Duan and Zhang [92] address the problem of deoccluding and recognizing face profiles with large-pose variations and occlusions through BoostGAN, which has a coarse-to-fine structure. In the coarse part, i.e., multi-occlusion frontal view generator, an encoder–decoder network is used for eliminating occlusion and producing multiple intermediate deoccluded faces. Subsequently, the coarse outputs are refined through a boosting network for photo-realistic and identity-preserved face generation. Consequently, the discriminator has a multi-input structure.

Since BoostGAN is a one-stage framework, it cannot handle de-occlusion and frontalization concurrently, which means that it loses the discriminative identity information. Furthermore, BoostGAN fails to employ the mask guided noise prior information. To address these, Duan et al. [93] perform face frontalization and face completion simultaneously. They propose an end-to-end mask guided two-stage GAN (TSGAN) framework. Each stage has its own generator and discriminator, while the first stage contains the face deocclusion module, the second one contains face frontalization module. Another module named mask-attention module (MAM) is deployed in both stages. The MAM encourages the face deocclusion module to concentrate more on missing regions and fills them based on the masked image input. The recovered image is fed into the second stage to obtain the final frontal image. TSGAN is trained with defined occlusion types and specified sizes, and multiple natural occlusions are not considered.

Table 3 provides an outline of the above-mentioned works, summarizing the type of GAN, the objective function, the dataset, and the results of each work.

**Table 3.** Summary of the face completion works, highlighting the types of GAN, loss functions, and the datasets that were used. The results are the reported results of the quality of the generated images (except for the ones marked with †). Results with * are the mean value of the published results. AL: Adversarial loss. PL: Perceptual loss. RL: Reconsturction loss. PSNR: Peak Signal-to-Noise Ratio ↑. SSIM: Structural Similarity ↑. ID: Identity Distance ↓. DIR@FAIR: Detection and Identification Rate at False Positive Rates ↑. MSE: Mean Square Error ↓. IS: Inception Score ↑. FID: Frechet Inception Distance ↓. NRMSE: Normalized Root MSE ↓.

| # | Paper | Type of GAN | Loss Function | Dataset | Results |
|---|---|---|---|---|---|
| 1. | Cai et al. [79] | OA-GAN | 1. Training with synthetic occlusion: PL, style loss, pixel loss, smoothness loss, $\mathcal{L}2$ loss, and AL. 2. Training with natural images: smoothness loss, $\mathcal{L}2$ loss, and AL. | CelebA [94] | PSNR = 22.61, SSIM = 0.787 |
| 2. | Chen et al. [80] | DCGAN | AL. | LFW [95] | Equal Error Rate (EER) $^{*\dagger}$ = 0.88 |
| 3. | Cheung et al. [81] | FSG-GAN | $\mathcal{L}1$ loss, identity-preserve loss, and AL. | CelebA, and LFW. | CelebA: PSNR * = 20.7513, SSIM = 0.8318; LFW: PSNR * = 20.8905, SSIM = 0.8527 |
| 4. | Li et al. [82] | GAN with two discriminators | Local and global AL, RL ($\mathcal{L}2$), and pixel-wise softmax loss. | CelebA, and Helen [96]. | PSNR * = 19.60, SSIM * = 0.803, ID = 0.470 |
| 5. | Mathai et al. [83] | GAN (modified generator) with two discriminators | RL ($\mathcal{L}1$), global WGAN loss, and local PatchGAN loss. | 1. Training the inpainter: CASIA WebFaces [97], VGG Faces [98], and MS-Celeb-1M [99]. 2. Testing the model: LFW, and LFW-BLUFR [100]. | DIR@FAR $^{\dagger}$ = 89.68 |
| 6. | Liu et al. [84] | GAN (modified generator) with two discriminators | RL ($\mathcal{L}2$), and AL. | CelebAMask-HQ [101] | NRMSE $^{\dagger}$ = 6.96 (result of facial landmark detection) |
| 7. | Cai et al. [85] | FCSR-GAN | MSE loss, PL, local and global AL, and face parsing loss. | CelebA, and Helen. | CelebA: PSNR = 20.22, SSIM = 0.780; Helen: PSNR = 20.01, SSIM = 0.761 |
| 8. | Li et al. [87] | GAN (modified generator) with two discriminators | Local and global AL, RL, and contextual attention loss. | CelebA, AR [102], and LFW. | Recognition accuracy $^{\dagger}$ = 95.44% |
| 9. | Dong et al. [88] | OA-GAN | AL and $\mathcal{L}1$ loss | CelebA, and CK+ [103], with additional occlusion images from the Internet. | PSNR * = 22.402, SSIM * = 0.753 |
| 10. | Jabbar et al. [89] | AFD-StackGAN (PatchGAN discriminators) | $\mathcal{L}1$ loss, RL ($\mathcal{L}1$, and SSIM), PL, and AL. | Custom dataset. | PSNR = 33.201, SSIM = 0.978, MSE = 32.435, NIQE (↓) = 4.902, BRISQUE (↓) = 39.872 |
| 11. | Li et al. [90] | DF-GAN | AL and cycle loss. | AR, Multi-PIE [104], Color FERET [105], and LFW. | AR: PSNR = 23.85, SSIM = 0.9168; MultiPIE: PSNR = 28.21, SSIM = 0.9176; FERET: PSNR = 28.15, SSIM = 0.931; LFW: PSNR = 23.18, SSIM = 0.869 |
| 12. | Jabbar et al. [91] | FD-StackGAN | RL ($\mathcal{L}1$, and SSIM loss), PL, and AL. | Custom dataset. | PSNR = 32.803, SSIM = 0.981, MSE = 34.145, NIQE (↓) = 4.499, BRISQUE (↓) = 42.504 |
| 13. | Duan and Zhang. [92] | BoostGAN | AL, identity preserving loss, $\mathcal{L}1$ loss, symmetry loss, and total variation (TV) loss. | Multi-PIE and LFW. | Recognition rate $^{\dagger}$ = 96.02 |
| 14. | Duan et al. [93] | TSGAN | AL, dual triplet loss, $\mathcal{L}1$ loss, symmetry loss, and TV loss. | Multi-PIE and LFW. | Recognition rate $^{\dagger}$ = 96.87 |
| 15. | Cong and Zhou. [106] | DCGAN | Cycle consistency loss from CycleGAN, AL, and Wasserstain distance loss. | Wider Face [107]. | IS = 10.36; FID = 8.85 |

#### 5.3.3. Attribute Classification

With the availability of surveillance cameras, the task of object detection and tracking through its visual appearance in a surveillance footage has gained prominence. Furthermore, there are other characteristics of people that are essential to fully understand an observed scene. The task of recognizing the people attributes (age, sex, race, etc.) and the items they hold (backpacks, bags, phone, etc.) is called attribute classification.

However, occluding the person in question by another person may lead to incorrectly classifying the attributes of the occluder instead of the occludee. Furthermore, the quality of the images from the surveillance cameras is usually low. Therefore, Fabbri et al. [108] focus on the poor resolution and occlusion challenges in recognizing the attribute of people such as gender, race, clothing, etc., in surveillance systems. The authors propose a model based on DCGAN [109] to improve the quality of images in order to overcome the mentioned problems. The model has three networks, one for attribute classification from the full body images, and the other two networks attempt to enhance the resolution and recover from occlusion. Eliminating the occlusion produces an image without noise and the residual of other subjects that could result in misclassification. However, under severe occlusions, the reconstructed image still contains the remaining of the occluder and the model fails to keep the parts of the image that should stay unmodified.

Similarly, Fulgeri et al. [110] tackle the occlusion issue by implementing a combination of UNet and GAN architecture. The model requires as input the occluded person image and its corresponding attributes. The generator takes the input and restores the image. The output is then forwarded to three networks: ResNet-101 [77], VGG-16 [111], and the discriminator to calculate the loss. The loss is backpropagated to update the weights of the generator. The goal of the model is to obtain a result image of a person that (a) is not occluded, (b) is similar at the pixel level to a person shape, and (c) contains the similar visual features as the original image. The results show that the model can detect and remove occlusion without any additional information. However, the model fails to fully recover the pixels around the boundary of the body parts. The authors constraint the input images by not having occlusion of more than six-sevenths of the image height.

#### 5.3.4. Miscellaneous Applications

In this section, we present the applications of GAN for amodal content completion in various categories of data.

**Food:** Papadopoulos et al. [112] present a compositional layer-based generative network called PizzaGAN that follows the steps of a recipe to make a pizza. The framework contains a pair of modules to add and remove all instances of each recipe component. A Cycle-GAN [6] is used to design each module. In the case of adding an element to the existing image, the module produces the appearance and the mask of the visible pixels in the new layer. Moreover, the removal module learns how to fill the holes that are left from the erased layer and generate the mask of the removed pixels. However, the authors do not provide any quantitative assessment of PizzaGAN.

**Vehicles:** Yan et al. [67] propose a two-part model to recover the amodal mask of a vehicle and the appearance of its hidden regions iteratively. To tackle both tasks, the model is composed of two parts: a segmentation completion module and an appearance recovery module. The first network is to complement the segmentation mask of the vehicle's invisible region. In order to complete the content of the occluded region, the appearance recovery module has a generator with a two-path network structure. The first path accepts the input image, the recovered mask from the segmentation completion module, and the modal mask, while learning how to fill in the colors of the hidden pixels. The other path requires the recovered mask and the ground-truth complete mask and learns how to use the image context to inpaint the whole foreground vehicle. The two paths share parameters, which increases the ability of the generator. To enhance the quality of the recovered image, it is taken through the whole model several times. However, the performance of the model degrades beyond three iterations for real images if occlusions are not severe.

**Humans:** The process of matching the same person in images taken by multiple cameras is referred to as Person re-identification (ReID). In surveillance systems where the purpose is to track and identify the individuals, ReID is essential. However, the stored images usually have low resolution and are blurry because they are from ordinary surveillance cameras [113]. Additionally, occlusion by other individuals and/or objects is most likely to occur since each camera has a different angle of view. Hence, some important features become difficult to recognize.

To tackle the challenge of person re-identification under occlusion, Tagore et al. [114] design a bi-network architecture with an Occlusion Handling GAN (OHGAN) module. An image with synthetic added occlusion is fed into the generator which is based on UNet architecture and produces an occlusion-free image by learning a non-linear project mapping function between the input image and the output image. Afterward, the discriminator computes the metric difference between the generated image and the original one. The ablation studies for the reconstruction task illustrate that the quality of completion is good for 10–20% occlusion and average for 30–40% occlusion. However, the quality of reconstruction degrades for occlusions higher than 50%.

On the other hand, Zhang et al. [66] attempt to complete the mask and the appearance of an occluded human through a two-stage network. First, the amodal completion stage predicts the amodal mask of the occluded person. Afterward, the content recovery network completes the RGB appearance of the invisible area. The latter uses a UNet architecture in the generator, with local and global discriminators to ensure that the output image is consistent with the global semantics while enhancing the clarity and contrast of the local regions. The generator adds a Visible Guided Attention (VGA) module to the skip connections. The VGA module computes a relational feature map to guide the low-level features to complete by concatenating the high-level features with the next-level features. The relational feature map represents the relation between the pixels inside and outside the occluded area. The process of extracting feature maps is similar to the self-attention mechanism in SAGAN by Zhang et al. [54]. Although incorporating VGA leads to a more accurate recovery of the content and texture, the model does not perform well on real images as it does on synthetic images.

### *5.4. Training Data*

Supervised learning frameworks require annotated ground-truth data to train a model. These data can be either from a manually annotated dataset, a synthetic occluded data from 3D computer-generated images, or by superimposing a part of an object/image on another object. For example, Ehsani et al. [76] train their model (SeGAN) on a photo-realistic synthetic dataset, and Zhan et al. [75] apply a self-supervised approach to generate annotated training data. However, a model trained with synthetic data may fail when it is tested on real-world data, and human-labeled data are costly, time-consuming, and susceptible to subjective judgments.

In this section, we discuss how GAN is implemented to generate training data for several categories.

**Generic objects:** It is nearly impossible to cover all the probable occlusions, and the likelihood of appearance of some occlusion cases is rather small. Therefore, Wang et al. [115] aim to utilize the data to improve the performance of the object detection in the case of occlusions. They utilize an adversarial network to generate hard examples with occlusions, and use them to train a Fast-RCNN [116]. Consequently, the detector becomes invariant to occlusions and deformations. Their model contains an Adversarial Spatial Dropout Network (ASDN), which takes as input features from an image patch and predicts a dropout mask that is used to create occlusion such that it would be difficult for Fast-RCNN to classify.

Likewise, Han et al. [117] apply an adversarial network to produce occluded adversary samples to train an object detector. The model, named Feature Fusion and Adversary Networks (FFAN), is based on Faster RCNN [118] and consists of a feature fusion network and an adversary occlusion network, and while the feature fusion module produces a feature map of high resolution and high semantic information to detect small objects more effectively, the adversary occlusion module produces occlusion on the feature map of the object thus outputs an adversary training sample that would be hard for the detector to discriminate. Meanwhile, the detector becomes better in classifying the generated occluded adversary samples through self-learning. Over time, the detector and the adversary occlusion network learn and compete with each other to enhance the performance of the model.

The occlusions produced by adversary networks in [115,117] may lead to overgeneralization, because they are similar to other class instances. For example, the occluded wheels of a bicycle results in misclassifying a wheel chair as a bike.

**Humans:** Zhao et al. [119] augment the input data to produce easy-to-hard occluded samples with different sizes and positions of the occlusion mask to increase the variation of occlusion patterns. They address the issue of ReID under occlusion through an Incremental Generative Occlusion Adversarial Suppression (IGOAS) framework. The network contains two modules, an incremental generative occlusion (IGO) block, and a global adversarial suppression (G&A) module. IGO takes the input data through augmentation and generates easy occluded samples. Then, it progressively enlarges the size of the occlusion mask with the number of training iterations. Thus, the model becomes more robust against occlusion as it learns harder occlusion incrementally rather than hardest ones directly. On the other hand, G&A consists of a global branch which extracts global features of the input data, and an adversarial suppression branch that weakens the response of the occluded region to zero and strengthens the response to non-occluded areas.

Furthermore, to increase the number of samples per identity for person ReID, Wu et al. [120] use a GAN network to synthesize labeled occluded data. Specifically, the authors impose block rectangles on the images to create random occlusion on the original person images which the model then tries to complete. The completed images that are similar but not identical to the original input are labeled with the same annotation as the corresponding raw image. Similarly, Zhang et al. [113] follow the same strategy to expand the original training set, expect that an additional noise channel is applied on the generated data to adjust the label further. Both approaches in [113,120] work with rectangular masks, but in real-world examples occlusions appear in free-form shapes.

**Face images:** Cong and Zhou [106] propose an improved GAN to generate occluded face images. The model is based on DCGAN with an added S-coder. The purpose of the S-coder is to force the generator to produce multi-class target images. The network is further optimized through Wasserstein distance and the cycle consistency loss from CycleGAN. However, only sunglasses and facial masks are considered as occlusive elements.

Figure 4 outlines of the discussed approaches for tackling the issues in overcoming occlusion through using GAN. Table 4 summarizes the GAN model, the loss function, and the datasets that were used in the discussed works in this section (except for the face completion works), it also shows the reported result for the tasks where GAN was applied.

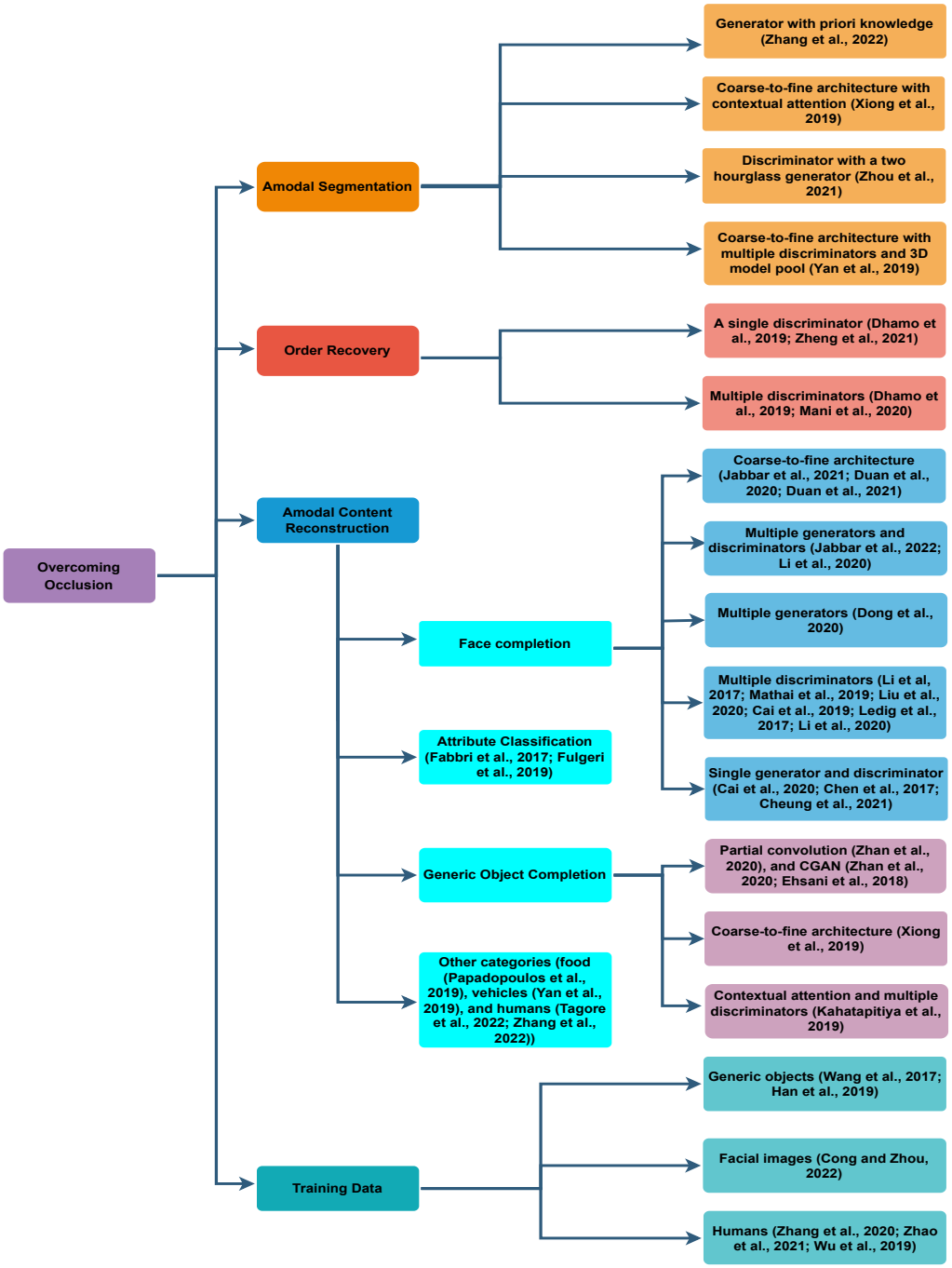


**Figure 4.** Outline of the approaches for addressing the challenges in overcoming occlusion through GAN. For amodal segmentation the implemented architecture are, a discriminator with a two hourglass generator [60], a coarse-to-fine architecture with contextual attention [63] or multiple discriminators [67], and a generator with priori knowledge [66]. For order recovery, GAN is designed as a generator with a single discriminator [71,73], or multiple discriminators [69,70]. To perform amodal content completion for facial images, the architectures include: a single generator and discriminator [79–81], multiple discriminators [82–87], multiple generators [88], multiple generators and discriminators [89,90], or a coarse-to-fine architecture [91–93]. Generic object completion is carried out through coarse-to-fine architecture [63], multiple discriminators with contextual attention [78], or partial convolution and CGAN [75,76]. Human completion for attribute classification is utilized in [108,110]. Other works use GAN to complete the images of food [112], vehicles [67], and humans [66,114]. GAN is also used to generate training data of generic objects [115,117], humans [113,119,120], and face images [106].

**Table 4.** Summary of the discussed works, highlighting the type of GAN, loss function, and the datasets that were used. The last two columns show the task that GAN was utilized for and its corresponding reported results. IoU: Intersection over Union ↑, R: Recall ↑, P: Precision ↑, ICP: Inception Conditional Probability ↑, SS: Segmentation Score ↑, Rel: Relative error ↓, RMSE: Root MSE ↓, MPE: Mean Pixel Error ↓, mIoU: mean IoU ↑, mAP: mean Average Precision ↑, mA: mean Accuracy ↑, AP: Average Precision ↑.

| # | Paper | Model | Loss Function | Dataset | Task | Results |
|---|---|---|---|---|---|---|
| 1. | Zhou et al. [60] | GAN with PGA | 1. For mask generation: binary cross-entropy (BCE), adversarial loss, and $\mathcal{L}1$ loss. 2. For content completion: adversarial loss, $\mathcal{L}1$ loss, perceptual loss, and style loss. | AHP (custom dataset) | Amodal segmentation and content completion | For mask generation: IoU = 86.1/40.3, L1 = 0.1635; For content completion: FID = 19.49, L1 = 0.0617 |
| 2. | Xiong et al. [63] | Coarse-to-fine structure with a PatchGAN discriminator | 1. For contour completion: a focal loss based content loss, and Hinge loss for adversarial loss. 2. For content completion: $\mathcal{L}1$ loss. | Places2 [121], and custom-designed dataset | Contour and content completion | L1 = 0.009327, L2 = 0.002329, PSNR = 29.86, SSIM = 0.9383, user study = 731 out of 1099 valid votes |
| 3. | Zhang et al. [66] | GAN with multiple PatchGAN discriminators | 1. For mask generation: adversarial loss, perceptual loss, and BCE loss. 2. For content generation: adversarial loss, $\mathcal{L}1$ loss, style loss, content loss, and TV loss. | Custom dataset | Amodal segmentation and content completion | For mask generation: mIoU = 0.82, L1 = 0.0638; For content completion: L1 = 0.0344, L2 = 0.0324, FID = 33.28 |
| 4. | Yan et al. [67] | GAN with multiple PatchGAN-based discriminators | $\mathcal{L}1$ loss, perceptual loss, and adversarial loss. | OVD (custom dataset) | Amodal segmentation and content completion | For mask generation: P = 0.9854, R = 0.8148, F1 = 0.8898, IoU = 0.8066, L1 = 0.0320, L2 = 0.0314; For content completion: ICP = 0.8350, SS = 0.9356, L1 = 0.0173, L2 = 0.0063 |
| 5. | Dhamo et al. [69] | PatchGAN-based | Adversarial loss and $\mathcal{L}1$ loss | SceneNet [122] and NYU depth v2 [123] | RGB-D completion | Rel = 0.017, RMSE = 0.095, SSIM = 0.903, RMSE = 19.76, PSNR = 22.22 |
| 6. | Dhamo et al. [73] | Original GAN | 1. For object completion: $\mathcal{L}1$ loss 2. For layout prediction: reconstruction ($\mathcal{L}1$) loss, perceptual loss, and adversarial loss. | SunCG [124] and Stanford2D-3D [125] | RGBA-D completion | SunCG: MPE = 43.12, RMSE = 65.66; Stanford2D-3D: MPE = 42.45, RMSE = 54.92 |
| 7. | Mani et al. [70] | GAN with two discriminators | $\mathcal{L}2$ loss, adversarial loss, and the discriminator loss. | KITTI [126] and Argoverse [127] | Scene completion | KITTI object: mIoU = 26.08, mAP = 40.79; KITTI tracking: mIoU = 24.16, mAP = 36.83; Argoverse: mIoU = 32.05, mAP = 48.31 |
| 8. | Zheng et al. [71] | GAN with two discriminators | Reconstruction loss, adversarial loss, and perceptual ($\mathcal{L}1$) loss. | COCOA[128], KINS [129], and CSD (custom dataset) | Scene completion | RMSE = 0.0914, SSIM = 0.8768, PSNR = 30.45 |
| 9. | Zhan et al. [75] | PCNet with CGAN | 1. For mask generation: BCE loss. 2. For content completion: losses in PC [62], $\mathcal{L}1$ loss, perceptual loss, and adversarial loss. | COCOA, and KINS | Content completion | KINS: mIoU = 94.76%; COCOA: mIoU = 81.35% |
| 10. | Ehsani et al. [76] | SeGAN | 1. For mask generation: BCE loss. 2. For content generation: Adversarial loss and $\mathcal{L}1$ loss. | DYCE (custom dataset) | Content completion | L1 = 0.07, L2 = 0.03, user study = 69.78% |

**Table 4.** *Cont.*

| # | Paper | Model | Loss Function | Dataset | Task | Results |
|---|---|---|---|---|---|---|
| 11. | Kahatapitiya et al. [78] | Inpainter with contextual attention | Spatially discounted reconstruction $\mathcal{L}1$ loss, local and global WGAN-GP adversarial loss. | COCO-Stuff [130] and MS COCO [131] | Content completion | User study positive = 79.7%, negative = 20.3% |
| 12. | Fabbri et al. [108] | DCGAN-based | 1. For attribute classification: weighted BCE loss. 2. For content completion: reconstruction loss and adversarial loss of the generator. | RAP [132] | Content completion | mA = 65.82, accuracy = 76.01, P = 48.98, R = 55.50, F1 = 52.04 |
| 13. | Fulgeri et al. [110] | Modified GAN (one generator and three discriminators) | Adversarial loss, content loss, and attribute loss (weighted BCE). | RAP, and Aic (custom dataset) | Content completion | RAP: mA = 72.18, accuracy = 59.59, P = 73.51, R = 73.72, F1 = 73.62, SSIM = 0.8239, PSNR = 20.65; AiC: mA = 78.37, accuracy = 53.3, P = 55.73, R = 85.46, F1 = 67.46, SSIM = 0.7101, PSNR = 21.81 |
| 14. | Papadopoulos et al. [112] | PizzaGAN | Adversarial loss, classification loss, cycle consistency loss as in CycleGAN, mask regularization mask. | Custom dataset | Amodal segmentation and content completion | Mask generation mIoU = 29.30% (quantitative results are not reported for content generation) |
| 15. | Zhang et al. [113] | CGAN | Adversarial loss. | Market-1501 [133] | Content completion | mAP = 90.42, Rank-1 = 93.35, Rank-5 = 96.87, Rank-10 = 97.92 |
| 16. | Tagore et al. [114] | OHGAN | BCE loss, and $\mathcal{L}2$ loss. | CUHK01 [134], CUHK03 [135], Market-1501, and DukeMTMC-reID [136] | Content completion | CUHK01: Rank-1 = 93.4, Rank-5 = 96.4, Rank-10 = 98.8; CUHK03: Rank-1 = 92.8, Rank-5 = 95.4, Rank-10 = 97.0; Market-1501: Rank-1 = 94.0, Rank-5 = 96.4, Rank-10 = 97.5, mAP = 86.4; DukeMTMC-reID: Rank-1 = 91.2, Rank-5 = 93.4, Rank-10 = 95.8, mAP = 82.4 |
| 17. | Wang et al. [115] | A custom-designed adversarial network | BCE loss. | VOC2007, VOC2012 [137], and MS COCO | Occlusion generation and deformation | VOC2007: mAP = 73.6; VOC2012: mAP = 69.0; MS COCO: $AP^{50}$ = 27.1 |
| 18. | Han et al. [117] | Adversary occlusion module | BCE loss. | VOC2007, VOC2012, MS COCO, and KITTI | Occlusion generation | VOC2007: mAP = 78.1; VOC2012: mAP = 76.7; MS COCO: AP = 42.7; KITTI: mAP = 89.01 |
| 19. | Wu et al. [120] | Original GAN | Euclidean loss, and BCE loss. | CUHK03, Market-1501, and DukeMTMC-reID | Content completion | Market-1501: mAP = 90.36, Rank-1 = 93.29, Rank-5 = 96.96, Rank-10 = 97.68; DukeMTMC-reID: mAP = 82.81, Rank-1 = 86.35, Rank-5 = 92.87, Rank-10 = 94.56; CUHK03: mAP = 61.95, Rank-1 = 59.78, Rank-5 = 70.64 |

## 6. Loss Functions

In GAN, the generator $G$ and the discriminator $D$ play against each other in a two-player mini–max game until they reach Nash equilibrium through a gradient-based optimization method. The gradient of the loss value indicates the learning performance of the network. The loss value is calculated via a loss (objective) function. In fact, defining a loss function is one the fundamental elements of designing GAN. Consequently, numerous objective functions have been proposed to stabilize and regularize GAN. The following losses are the most common ones used in training GAN for amodal completion.

1. **Adversarial Loss**: The loss function used in training GAN is known as an adversarial loss. It measures the distance between the distribution of the generated sample and the real sample. Each of $G$ and $D$ have their dedicated loss function which together form the adversarial loss, as shown in Equation (1). However, $G$ is trained as the term that reflects the distribution of the generated data ($\mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z)))]$). Extensions to the original loss function are the conditional loss and the Wasserstein loss defined in CGAN and WGAN, respectively.
2. **Content Loss**: In image generation, content loss [138] measures the difference between the content representation of the real and the generated images, to make them more similar in terms of perceptual content. If $p$ and $x$ are the original and the generated images, and $p^l$ and $X^l$ are their respective representations in layer $l$, the content loss is calculated as
$$\mathcal{L}_{content}(p,x,l) = \frac{1}{2}\sum_{i,j}(F_{ij}^l - P_{ij}^l)^2 \tag{3}$$
3. **Reconstruction Loss**: The key idea behind reconstruction loss proposed by Li et al. [139] is to benefit from the visual features learned by $D$ from the training data. The extracted features from the real data by $D$ are fed to $G$ to regenerate real data. By adding reconstruction loss to the GAN's loss function, $G$ is encouraged to reconstruct from the features of $D$, which brings $G$ closer to the configurations of the real data. The reconstruction loss equation is as follows:
$$\mathcal{L}_X^{\phi,\theta} = \mathbb{E}_{x\sim p_x}[\|G^\theta(D_F^\phi(X)) - X\|_1] \tag{4}$$
where $D_F^\phi$ is a part of the discriminator which encodes the data to features, and $G^\theta$ decodes the features to the training data.
4. **Style Loss**: The style loss, originally designed for image style transfer by Gatys et al. [138], is defined to ensure that the style representation of the generated image matches that of the input style image. It depends on the feature correlation between the feature maps, given by the Gram matrix ($G^l$). Let $a$ and $x$ be the original image and the generated image, respectively, and $A^l$ and $G^l$ their corresponding style representation in layer $l$. The style loss is computed by the element-wise mean square difference between $A^l$ and $G^l$,
$$\mathcal{L}_{style}(a,x) = \sum_{l=0}^{L} w_l \frac{1}{4N_l^2M_l^2}\sum_{i,j}(G_{ij}^l - A_{ij}^l)^2 \tag{5}$$
where $w^l$ is the weighting factor of each layer, and $N$ and $M$ represent the number and the size of the feature maps, respectively.
5. **$\mathcal{L}_1$ and $\mathcal{L}_2$ Loss**: $\mathcal{L}_1$ loss function is the absolute difference between the ground-truth and the generated image. On the other hand, $\mathcal{L}_2$ loss is the squared difference between the actual and the generated data. When used alone, these loss functions lead to blurred results [140]. However, when combined with other loss functions, they can improve the quality of the generated images, especially $\mathcal{L}_1$ loss. The generator is encouraged to not only fool the discriminator but also to be closer to the real data in $\mathcal{L}_1$ or $\mathcal{L}_2$ sense. Although these losses cannot capture high-frequency details, they

accurately capture low frequencies. $\mathcal{L}_1$ loss enforces correctness in low-frequency features; hence, it results in less blurred images compared to $\mathcal{L}_2$ [8]. Both losses are defined in Equations (6) and (7).

$$\mathcal{L}_1 = \mathbb{E}_{x,y,z}[\|y - G(x,z)\|_1] \tag{6}$$

$$\mathcal{L}_2 = \mathbb{E}_{x,y,z}[\|y - G(x,z)\|_2^2] \tag{7}$$

where $x$, $y$, and $z$ are the ground-truth image, the generated image, and the random noise, respectively.

6. **Perceptual Loss**: The perceptual loss measures the high-level perceptual and semantic differences between the real and the fake images. Several works [141,142] introduce perceptual loss as a combination of the content loss (or feature reconstruction loss) and the style loss. However, Liu et al. [62] simply compute the $L_1$ distance between the real and the completed images. Others incorporate more similarity metrics into it [140].
7. **BCE Loss**: BCE loss measures how close the probability of the predicted data is to the real data. Its value increases as the predicted probability deviates from the real label. The BCE is defined as

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}(y_i log(D(i) + (1 - y_i) log(1 - D(i))) \tag{8}$$

where $y_i$ is the label of $i$. $y_i$=0 and $y_i$=1 represents fake and real samples.
BCE is used in training the discriminator in amodal segmentation task [76], and in training the generator [110].
8. **Hinge Loss**: In GAN, Hinge loss is used to help the convergence to a Nash equilibrium. Proposed by Lim and Ye [143], the objective function for $G$ is

$$\mathcal{L}_G = -\mathbb{E}_{z\sim p_z(z)}[D(G(z))] \tag{9}$$

and for $D$ is

$$\mathcal{L}_D = \mathbb{E}_{x\sim p_{data}(x)}[max(0, 1 - D(x)] + \mathbb{E}_{z\sim p_z(z)}[max(0, 1 + D(G(z))] \tag{10}$$

where $x$ and $z$ are the ground-truth and the generated images, respectively.

As it can be seen from Tables 3 and 4, many of the previously mentioned loss functions are combined with others to train a GAN model. Adversarial loss is the base objective function for training the two networks of the GAN. However, with the original GAN's adversarial loss function, the model may not converge. Therefore, the Hinge loss is often implemented as an alternative objective function. In some works, global and local adversarial losses are used to train local and global discriminators to ensure that the generated data is semantically and locally coherent. In addition to this, $\mathcal{L}_1$ or $\mathcal{L}_2$ losses are frequently utilized to capture low-frequency features, and hence improve the quality of the generated images. Furthermore, the reconstruction loss is employed to encourage the generator to maintain the contents of the original input image. On the other hand, perceptual loss encourages the model to capture patch-level information when completing a missing patch in an object/image. Furthermore, to emphasize on the style match between the generated image and the input image, style loss is implemented.

The choice of the objective functions is an essential decision of designing a model. In amodal completion and inpainting, designing a loss function is still an active area of research. The ablation studies performed by the reviewed works show that there is no optimal objective function. For different tasks and data, a different set of loss terms produces the best results. In addition, using a complex loss function may lead to problems of instability, vanishing gradient, and mode collapse.

## 7. Open Challenges and Future Directions

Despite the significant progress of the research in GAN and amodal completion in the last decade, there remain a number of problems that can be considered as future directions.

1. **Amodal training data**: Up until now, there has been no fully annotated generic amodal dataset with sufficient ground-truth labels for the three sub-tasks in amodal completion. Most of the existing datasets are specific to a particular application or task. This not only makes training the models themselves more difficult, but verifying their learning capability as well. In many cases, there is no sufficient labeled amodal validation data to establish the accuracy of the model. We present the challenges related to each sub-task in amodal completion.
   For amodal segmentation, the current datasets do not contain sufficient occlusion cases between similar objects. Hence, the model cannot tell where the boundary of one object ends and the other one begins.
   The existing real (manually annotated) amodal datasets have no ground-truth appearance for the occluded region. This makes training and validating the model for amodal content completion more challenging.
   As for the case of order recovery, some occlusion situations are very rare in the existing datasets. On the other hand, it is impossible to cover all probable cases of occlusion in the real datasets. Nevertheless, in the future, the current datasets can be extended through generated occlusion to include more of those infrequent cases with varying degrees of occlusion.
2. **Evaluation metrics**: There are several quantitative and qualitative evaluation measures for GAN [59]. However, as it can be noticed from the results, there is no standard and unanimous evaluation metric for assessing the performance of GAN when it generates the occluded content. Many existing works depend on the human preference judgement which can be biased and subjective. Therefore, designing a consensus evaluation metric is of utmost importance.
3. **Reference data**: Existing GAN models fail to generate occluded content accurately if the hidden area is large. Particularly, when the occluded object is non-symmetric, such as the face or the human body. The visible region of the object may not hold sufficient relevant features to guide a visually plausible regeneration. As the next step, reference images can be used along the input image to guide the completion more effectively.

In addition to the above-mentioned problems, the challenges in the stability and convergence of GAN remain open issues [28].

## 8. Discussion

Current computational models approach the human capability of visible perception when performing visual tasks such as recognition, detection, and segmentation. However, our environment is complex and dynamic. Most of the objects we perceive are incomplete and fragmented. Therefore, the existing models that are designed and trained with a fully visible sample of instances do not perform well when tested on real-world scenes. Hence, overcoming occlusion is essential for leveraging the performance of available models. Amodal completion tasks address the occluded patches of an image to infer the occlusion relation between objects (i.e., order recovery), predict the full shape of the objects (i.e., amodal segmentation), and complete the RGB appearance of the missing pixels (i.e., amodal content completion). These tasks are usually interleaved and depend on each other. For example, amodal segmentation can benefit order recovery [144] and it is crucial for amodal content completion [76]. On the other hand, order recovery can guide the amodal segmentation [75].

Although GAN is notorious for its stability issues and is difficult to train, it is a popular approach for tasks that require generative capability. In handling occlusion, the initially incomplete representation needs to be extended to a complete representation with the miss-

ing region filled in. Therefore, GAN is the chosen architecture for processes/sub-processes involved in amodal completion. However, depending on the nature of the problems, the applicability of GAN varies. For example, in amodal appearance reconstruction, GAN is the ideal option of architecture and it produces superior results in comparison to other methods. Comparably, in amodal segmentation and order recovery tasks, GAN is less commonly used. Nevertheless, to take advantage of the potential of GAN, it can be combined with other architectures and learning strategies to tackle those tasks too.

In order to help GAN in learning implicit features from the visible regions of the image, various methods are used, which can be summarized as follows:

- **Architecture:** While the original GAN consists of a single generator and discriminator, several works utilize multiple generators and discriminators. The implementation of local and global discriminators is especially common, because it enhances the quality of the generated data. The generator is encouraged to concentrate on both the global contextual and local features, and produce images that are closer to the distribution of the real data. In addition to this, an initial-to-refined (also called coarse-to-fine) architecture is implemented in many models. The initial stage produces a coarse output from the input image, which is then further refined in the refinement step.
- **Objective function:** To improve the quality of the generated output and stabilize the training of the GAN, a combination of loss terms is used. While adversarial loss and Hinge loss are used in training the two networks in the GAN, other objective functions encourage the model to produce an image that is consistent with the ground-truth image.
- **Input:** Under severe occlusion, the GAN may fail to produce a visually pleasing output solely depending on the visible region. Therefore, providing additional input information guides GAN in producing better results. In the amodal shape and content completion, synthetic instances similar to the occluded object are useful, because they can be used as a reference by the model. A priori knowledge is also beneficial, as it can either be manually encoded (e.g., utilizing various human poses for human deocclusion) or transferred from a pre-trained model (e.g., using a pre-trained face recognition model in face deocclusion). In addition to these, employing the amodal mask and the category of the occluded object in the content completion task restricts the GAN model to focus on completing the object in question. For producing the amodal mask, a modal mask is needed as an input. If the input is not available, most of existing works depend on a pre-trained segmentation model to predict the visible segmentation mask.
- **Feature extraction:** The pixels in the visible region of an image are rather important and contain essential information for various tasks; hence, they are considered as valid pixels. Contrary to this, the invisible pixels are invalid ones; hence, they should not be included in the feature extraction/encoding process. However, the vanilla convolution process cannot differentiate between valid and invalid pixels, which generates images with visual artifacts and color discrepancies. Therefore, partial convolution and a soft gating mechanism are implemented to enforce the generator to focus only on valid pixels and eliminate/minimize the effect of the invalid ones. On the other hand, dilated convolution layers can replace the vanilla convolution layers to borrow information from relevant spatially distant pixels. Additionally, contextual attention layers and attention mechanism are added to the networks of the GAN to leverage the information from the image context and capture global dependencies.

Among the various architectures of GAN, three types are most commonly used in the reviewed works in this article, namely CGAN, WGAN-GP, and PatchGAN. The application of CGAN is mostly in amodal content completion tasks, because the GAN is encouraged to complete an object of a specific class. WGAN-GP stabilizes the training of GAN with an EM distance objective function and a weight clipping method. Therefore, it is a preferred architecture to ensure GAN convergence. On the other hand, PatchGAN is used in designing the discriminator, as it attempts to classify patches of the generated image as real or

fake. Consequently, the image is penalized for style consistency between pixels that are spatially more than a patch diameter away from each other.

Finally, handling occlusion is fundamental in several computer vision tasks. For example, completing an occluded facial image helps in better recognizing the face and predicting the identity of the person. Similarly, inferring the full shape of pedestrians and vehicles as well as the occlusion relationship between them can lead to a safer autonomous driving. Furthermore, in surveillance cameras, amodal completion helps in target tracking and security applications.

## 9. Conclusions

GANs are considered the most interesting idea in machine learning since their invention. Due to their generative capability, they are extending the ability of artificial intelligence systems. The GAN-based models are creative instead of mere learners. In the challenging field of amodal completion, GAN has had a significant impact especially in generating the appearance of a missing region. This brings existing vision systems closer to the human capability in predicting the occluded area.

To help the researchers in the field, in this survey we have reviewed the available works in the literature wherein a GAN is applied in accomplishing tasks of amodal completion and resolving the problems that arise when addressing occlusion. We discussed the architecture of each model along with its strengths and limitations in detail. Then, we summarized the loss function and the dataset that was used in each work and presented their results. Then, we discussed the most common types of objective functions which are implemented in training the GAN models for occlusion handling. Finally, we provided a discussion of the key findings of our survey article.

However, after reviewing the current progress in overcoming occlusion using a GAN, we detected several key issues that remain an open challenge in the research of addressing occlusion. These issues pave the way for the future research direction. By addressing them, the field will progress significantly.

**Author Contributions:** Conceptualization, K.S., S.S. and Z.V.; methodology, K.S. and S.S.; investigation and data curation, K.S.; writing—original draft preparation, K.S. and S.S.; writing—review and editing, K.S., S.S. and Z.V.; visualization, K.S.; supervision, S.S. and Z.V. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data will be made available upon request.

**Acknowledgments:** On behalf of the OHIOD project we are grateful for the possibility to use ELKH Cloud [145]. The authors would like to thank the High Performance Computing Research Group of Óbuda University for its valuable support. The authors also thank NVIDIA Corporation for their support.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Thielen, J.; Bosch, S.E.; van Leeuwen, T.M.; van Gerven, M.A.; van Lier, R. Neuroimaging findings on amodal completion: A review. *i-Perception* **2019**, *10*, 2041669519840047. [CrossRef] [PubMed]
2. Saleh, K.; Szénási, S.; Vámossy, Z. Occlusion Handling in Generic Object Detection: A Review. In Proceedings of the 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 21–23 January 2021; pp. 477–484.
3. Wang, Z.; She, Q.; Ward, T.E. Generative adversarial networks in computer vision: A survey and taxonomy. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–38. [CrossRef]
4. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. *arXiv* **2017**, arXiv:1710.10196.

5. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
6. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
7. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.
8. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
9. Yang, T.; Pan, Q.; Li, J.; Li, S.Z. Real-time multiple objects tracking with occlusion handling in dynamic scenes. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 21–23 September 2005; Volume 1, pp. 970–975.
10. Enzweiler, M.; Eigenstetter, A.; Schiele, B.; Gavrila, D.M. Multi-cue pedestrian classification with partial occlusion handling. In Proceedings of the 2010 IEEE Computer Society Conference on cOmputer Vision Furthermore, Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 990–997.
11. Benenson, R. Occlusion Handling. In *Computer Vision: A Reference Guide*; Ikeuchi, K., Ed.; Springer US: Boston, MA, USA, 2014; pp. 551–552. ._136. [CrossRef]
12. Tian, Y.; Guan, T.; Wang, C. Real-time occlusion handling in augmented reality based on an object tracking approach. *Sensors* **2010**, *10*, 2885–2900. [CrossRef]
13. Ao, J.; Ke, Q.; Ehinger, K.A. Image amodal completion: A survey. In *Computer Vision and Image Understanding*; Elsevier: Amsterdam, The Netherlands, 2023; p. 103661.
14. Anuj, L.; Krishna, M.G. Multiple camera based multiple object tracking under occlusion: A survey. In Proceedings of the 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bengaluru, India, 21–23 February 2017; pp. 432–437.
15. Shravya, A.; Monika, K.; Malagi, V.; Krishnan, R. A comprehensive survey on multi object tracking under occlusion in aerial image sequences. In Proceedings of the 2019 1st International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, India, 19–20 March 2019; pp. 225–230.
16. Ning, C.; Menglu, L.; Hao, Y.; Xueping, S.; Yunhong, L. Survey of pedestrian detection with occlusion. *Complex Intell. Syst.* **2021**, *7*, 577–587. [CrossRef]
17. Li, F.; Li, X.; Liu, Q.; Li, Z. Occlusion Handling and Multi-scale Pedestrian Detection Based on Deep Learning: A Review. *IEEE Access* **2022**, *10*, 19937–19957. [CrossRef]
18. Zhang, L.; Verma, B.; Tjondronegoro, D.; Chandran, V. Facial expression analysis under partial occlusion: A survey. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–49. [CrossRef]
19. Dagnes, N.; Vezzetti, E.; Marcolin, F.; Tornincasa, S. Occlusion detection and restoration techniques for 3D face recognition: A literature review. *Mach. Vis. Appl.* **2018**, *29*, 789–813. [CrossRef]
20. Zeng, D.; Veldhuis, R.; Spreeuwers, L. A survey of face recognition techniques under occlusion. *IET Biom.* **2021**, *10*, 581–606. [CrossRef]
21. Meena, M.K.; Meena, H.K. A Literature Survey of Face Recognition Under Different Occlusion Conditions. In Proceedings of the 2022 IEEE Region 10 Symposium (TENSYMP), Mumbai, India, 1–3 July 2022; pp. 1–6.
22. Biswas, S. Performance Improvement of Face Recognition Method and Application for the COVID-19 Pandemic. *Acta Polytech. Hung.* **2022**, *19*,1–21.
23. Gilroy, S.; Jones, E.; Glavin, M. Overcoming occlusion in the automotive environment—A review. *IEEE Trans. Intell. Transp. Syst.* **2019**, *22*, 23–35. [CrossRef]
24. Rosić, S.; Stamenković, D.; Banić, M.; Simonović, M.; Ristić-Durrant, D.; Ulianov, C. Analysis of the Safety Level of Obstacle Detection in Autonomous Railway Vehicles. *Acta Polytech. Hung.* **2022**, *1*, 187–205. [CrossRef]
25. Macedo, M.C.d.F.; Apolinario, A.L. Occlusion Handling in Augmented Reality: Past, Present and Future. *IEEE Trans. Vis. Comput. Graph.* **2021**, *29*, 1590–1609. [CrossRef]
26. Zhang, Z.; Ji, X.; Cui, X.; Ma, J. A Survey on Occluded Face recognition. In Proceedings of the 2020 The 9th International Conference on Networks, Communication and Computing, Tokyo, Japan, 18–20 December 2020; pp. 40–49.
27. Sajeeda, A.; Hossain, B.M. Exploring Generative Adversarial Networks and Adversarial Training. *Int. J. Cogn. Comput. Eng.* **2022**, *3*, 78–89. [CrossRef]
28. Saxena, D.; Cao, J. Generative adversarial networks (GANs) challenges, solutions, and future directions. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–42. [CrossRef]
29. Jabbar, A.; Li, X.; Omar, B. A survey on generative adversarial networks: Variants, applications, and training. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–49. [CrossRef]
30. Farajzadeh-Zanjani, M.; Razavi-Far, R.; Saif, M.; Palade, V. Generative Adversarial Networks: A Survey on Training, Variants, and Applications. In *Generative Adversarial Learning: Architectures and Applications*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 7–29.
31. Pavan Kumar, M.; Jayagopal, P. Generative adversarial networks: A survey on applications and challenges. *Int. J. Multimed. Inf. Retr.* **2021**, *10*, 1–24. [CrossRef]

32. Hong, Y.; Hwang, U.; Yoo, J.; Yoon, S. How generative adversarial networks and their variants work: An overview. *ACM COmputing Surv. (CSUR)* **2019**, *52*, 1–43. [CrossRef]
33. Li, Y.; Wang, Q.; Zhang, J.; Hu, L.; Ouyang, W. The theoretical research of generative adversarial networks: An overview. *Neurocomputing* **2021**, *435*, 26–41. [CrossRef]
34. Pan, Z.; Yu, W.; Yi, X.; Khan, A.; Yuan, F.; Zheng, Y. Recent progress on generative adversarial networks (GANs): A survey. *IEEE Access* **2019**, *7*, 36322–36333. [CrossRef]
35. Salehi, P.; Chalechale, A.; Taghizadeh, M. Generative adversarial networks (GANs): An overview of theoretical model, evaluation metrics, and recent developments. *arXiv* **2020**, arXiv:2005.13178.
36. Jin, L.; Tan, F.; Jiang, S. Generative adversarial network technologies and applications in computer vision. *Comput. Intell. Neurosci.* **2020**, *2020*, 1459107. [CrossRef] [PubMed]
37. Wang, K.; Gou, C.; Duan, Y.; Lin, Y.; Zheng, X.; Wang, F.Y. Generative adversarial networks: Introduction and outlook. *IEEE/CAA J. Autom. Sinica* **2017**, *4*, 588–598. [CrossRef]
38. Alotaibi, A. Deep generative adversarial networks for image-to-image translation: A review. *Symmetry* **2020**, *12*, 1705. [CrossRef]
39. Porkodi, S.; Sarada, V.; Maik, V.; Gurushankar, K. Generic image application using GANs (Generative Adversarial Networks): A Review. *Evol. Syst.* **2022**, 1–15. [CrossRef]
40. Kammoun, A.; Slama, R.; Tabia, H.; Ouni, T.; Abid, M. Generative Adversarial Networks for face generation: A survey. *ACM Comput. Surv. (CSUR)* **2022**, *55*, 1–37. [CrossRef]
41. Toshpulatov, M.; Lee, W.; Lee, S. Generative adversarial networks and their application to 3D face generation: A survey. *Image Vis. Comput.* **2021**, *108*, 104119. [CrossRef]
42. Tschuchnig, M.E.; Oostingh, G.J.; Gadermayr, M. Generative adversarial networks in digital pathology: A survey on trends and future potential. *Patterns* **2020**, *1*, 100089. [CrossRef]
43. Saad, M.M.; O'Reilly, R.; Rehmani, M.H. A Survey on Training Challenges in Generative Adversarial Networks for Biomedical Image Analysis. *arXiv* **2022**, arXiv:2201.07646.
44. Zhao, J.; Hou, X.; Pan, M.; Zhang, H. Attention-based generative adversarial network in medical imaging: A narrative review. *Comput. Biol. Med.* **2022**, *149*, 105948. [CrossRef]
45. Alqahtani, H.; Kavakli-Thorne, M.; Kumar, G. Applications of generative adversarial networks (gans): An updated review. *Arch. Comput. Methods Eng.* **2021**, *28*, 525–552. [CrossRef]
46. Sampath, V.; Maurtua, I.; Aguilar Martín, J.J.; Gutierrez, A. A survey on generative adversarial networks for imbalance problems in computer vision tasks. *J. Big Data* **2021**, *8*, 1–59. [CrossRef]
47. Ljubić, H.; Martinović, G.; Volarić, T. Augmenting data with generative adversarial networks: An overview. *Intell. Data Anal.* **2022**, *26*, 361–378. [CrossRef]
48. Tian, C.; Zhang, X.; Lin, J.C.W.; Zuo, W.; Zhang, Y.; Lin, C.W. Generative adversarial networks for image super-resolution: A survey. *arXiv* **2022**, arXiv:2204.13620.
49. Aggarwal, A.; Mittal, M.; Battineni, G. Generative adversarial network: An overview of theory and applications. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100004. [CrossRef]
50. Gui, J.; Sun, Z.; Wen, Y.; Tao, D.; Ye, J. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 3313–3332. [CrossRef]
51. Hitawala, S. Comparative study on generative adversarial networks. *arXiv* **2018**, arXiv:1801.04271.
52. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative adversarial networks: An overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [CrossRef]
53. Goodfellow Ian, J.; Jean, P.A.; Mehdi, M.; Bing, X.; David, W.F.; Sherjil, O.; Courville Aaron, C. Generative adversarial nets. In Proceedings of the 27th international Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 2, pp. 2672–2680.
54. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 7354–7363.
55. Rubner, Y.; Tomasi, C.; Guibas, L.J. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vis.* **2000**, *40*, 99–121. [CrossRef]
56. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of Wasserstein GANs. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5767–5777.
57. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training gans. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2234–2242.
58. Arjovsky, M.; Bottou, L. Towards principled methods for training generative adversarial networks. *arXiv* **2017**, arXiv:1701.04862.
59. Borji, A. Pros and cons of gan evaluation measures. *Comput. Vis. Image Underst.* **2019**, *179*, 41–65. [CrossRef]
60. Zhou, Q.; Wang, S.; Wang, Y.; Huang, Z.; Wang, X. Human De-occlusion: Invisible Perception and Recovery for Humans. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 3691–3701.
61. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.

62. Liu, G.; Reda, F.A.; Shih, K.J.; Wang, T.C.; Tao, A.; Catanzaro, B. Image inpainting for irregular holes using partial convolutions. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 14–17 May 2018; pp. 85–100.
63. Xiong, W.; Yu, J.; Lin, Z.; Yang, J.; Lu, X.; Barnes, C.; Luo, J. Foreground-aware image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5840–5848.
64. Rajchl, M.; Lee, M.C.; Oktay, O.; Kamnitsas, K.; Passerat-Palmbach, J.; Bai, W.; Damodaram, M.; Rutherford, M.A.; Hajnal, J.V.; Kainz, B.; et al. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE Trans. Med. Imaging* **2016**, *36*, 674–683. [CrossRef]
65. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Generative image inpainting with contextual attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5505–5514.
66. Zhang, Q.; Liang, Q.; Liang, H.; Yang, Y. Removal and Recovery of the Human Invisible Region. *Symmetry* **2022**, *14*, 531. [CrossRef]
67. Yan, X.; Wang, F.; Liu, W.; Yu, Y.; He, S.; Pan, J. Visualizing the invisible: Occluded vehicle segmentation and recovery. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7618–7627.
68. Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D.N. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 1947–1962. [CrossRef]
69. Dhamo, H.; Tateno, K.; Laina, I.; Navab, N.; Tombari, F. Peeking behind objects: Layered depth prediction from a single image. *Pattern Recognit. Lett.* **2019**, *125*, 333–340. [CrossRef]
70. Mani, K.; Daga, S.; Garg, S.; Narasimhan, S.S.; Krishna, M.; Jatavallabhula, K.M. Monolayout: Amodal scene layout from a single image. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1689–1697.
71. Zheng, C.; Dao, D.S.; Song, G.; Cham, T.J.; Cai, J. Visiting the Invisible: Layer-by-Layer Completed Scene Decomposition. *Int. J. Comput. Vis.* **2021**, *129*, 3195–3215. [CrossRef]
72. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
73. Dhamo, H.; Navab, N.; Tombari, F. Object-driven multi-layer scene decomposition from a single image. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5369–5378.
74. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Free-form image inpainting with gated convolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 4471–4480.
75. Zhan, X.; Pan, X.; Dai, B.; Liu, Z.; Lin, D.; Loy, C.C. Self-supervised scene de-occlusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3784–3792.
76. Ehsani, K.; Mottaghi, R.; Farhadi, A. Segan: Segmenting and generating the invisible. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6144–6153.
77. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
78. Kahatapitiya, K.; Tissera, D.; Rodrigo, R. Context-aware automatic occlusion removal. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1895–1899.
79. Cai, J.; Han, H.; Cui, J.; Chen, J.; Liu, L.; Zhou, S.K. Semi-supervised natural face de-occlusion. *IEEE Trans. Inf. Forensics Secur.* **2020**, *16*, 1044–1057. [CrossRef]
80. Chen, Y.A.; Chen, W.C.; Wei, C.P.; Wang, Y.C.F. Occlusion-aware face inpainting via generative adversarial networks. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1202–1206.
81. Cheung, Y.M.; Li, M.; Zou, R. Facial Structure Guided GAN for Identity-preserved Face Image De-occlusion. In Proceedings of the 2021 International Conference on Multimedia Retrieval, Taipei, Taiwan, 21 August 2021; pp. 46–54.
82. Li, Y.; Liu, S.; Yang, J.; Yang, M.H. Generative face completion. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3911–3919.
83. Mathai, J.; Masi, I.; AbdAlmageed, W. Does generative face completion help face recognition? In Proceedings of the 2019 International Conference on Biometrics (ICB), Crete, Greece, 4–7 June 2019; pp. 1–8.
84. Liu, H.; Zheng, W.; Xu, C.; Liu, T.; Zuo, M. Facial landmark detection using generative adversarial network combined with autoencoder for occlusion. *Math. Probl. Eng.* **2020**, *2020*, 1–8. [CrossRef]
85. Cai, J.; Hu, H.; Shan, S.; Chen, X. Fcsr-gan: End-to-end learning for joint face completion and super-resolution. In Proceedings of the 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), Lille, France, 14–18 May 2019; pp. 1–8.
86. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
87. Li, C.; Ge, S.; Zhang, D.; Li, J. Look through masks: Towards masked face recognition with de-occlusion distillation. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 3016–3024.

88. Dong, J.; Zhang, L.; Zhang, H.; Liu, W. Occlusion-aware gan for face de-occlusion in the wild. In Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME), London, UK, 6–10 July 2020; pp. 1–6.
89. Jabbar, A.; Li, X.; Assam, M.; Khan, J.A.; Obayya, M.; Alkhonaini, M.A.; Al-Wesabi, F.N.; Assad, M. AFD-StackGAN: Automatic Mask Generation Network for Face De-Occlusion Using StackGAN. *Sensors* **2022**, *22*, 1747. [CrossRef]
90. Li, Z.; Hu, Y.; He, R.; Sun, Z. Learning disentangling and fusing networks for face completion under structured occlusions. *Pattern Recognit.* **2020**, *99*, 107073. [CrossRef]
91. Jabbar, A.; Li, X.; Iqbal, M.M.; Malik, A.J. FD-StackGAN: Face De-occlusion Using Stacked Generative Adversarial Networks. *KSII TRansactions Internet Inf. Syst. (TIIS)* **2021**, *15*, 2547–2567.
92. Duan, Q.; Zhang, L. Look more into occlusion: Realistic face frontalization and recognition with boostgan. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 214–228. [CrossRef]
93. Duan, Q.; Zhang, L.; Gao, X. Simultaneous face completion and frontalization via mask guided two-stage GAN. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 3761–3773. [CrossRef]
94. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3730–3738.
95. Huang, G.B.; Mattar, M.; Berg, T.; Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In Proceedings of the Workshop on faces in 'Real-Life' Images: Detection, Alignment, and Recognition, Marseille, France, 17 October 2008.
96. Le, V.; Brandt, J.; Lin, Z.; Bourdev, L.; Huang, T.S. Interactive facial feature localization. In *Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 679–692.
97. Yi, D.; Lei, Z.; Liao, S.; Li, S.Z. Learning face representation from scratch. *arXiv* **2014**, arXiv:1411.7923.
98. Parkhi, O.M.; Vedaldi, A.; Zisserman, A. Deep Face Recognition In *Proceedings of the British Machine Vision Conference (BMVC), Swansea, UK, 7–10 September 2015*; BMVA Press, 2015; pp. 41.1–41.12.
99. Guo, Y.; Zhang, L.; Hu, Y.; He, X.; Gao, J. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 87–102.
100. Liao, S.; Lei, Z.; Yi, D.; Li, S.Z. A benchmark study of large-scale unconstrained face recognition. In Proceedings of the IEEE International Joint Conference on Biometrics, Clearwater, FL, USA, 29 September–2 October 2014; pp. 1–8.
101. Lee, C.H.; Liu, Z.; Wu, L.; Luo, P. Maskgan: Towards diverse and interactive facial image manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5549–5558.
102. Martinez, A.; Benavente, R. *The Ar Face Database: Cvc Technical Report, 24*; Universitat Autonoma do Barcelona: Barcelona, Spain, 1998.
103. Lucey, P.; Cohn, J.F.; Kanade, T.; Saragih, J.; Ambadar, Z.; Matthews, I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, San Francisco, CA, USA, 13–18 June 2010; pp. 94–101.
104. Gross, R.; Matthews, I.; Cohn, J.; Kanade, T.; Baker, S. Multi-pie. *IMage Vis. Comput.* **2010**, *28*, 807–813. [CrossRef]
105. Phillips, P.J.; Moon, H.; Rizvi, S.A.; Rauss, P.J. The FERET evaluation methodology for face-recognition algorithms. *IEEE TRansactions Pattern Anal. Mach. Intell.* **2000**, *22*, 1090–1104. [CrossRef]
106. Cong, K.; Zhou, M. Face Dataset Augmentation with Generative Adversarial Network. *J. Phys. Conf. Ser.* **2022**, *2218*, 012035. [CrossRef]
107. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. Wider face: A face detection benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5525–5533.
108. Fabbri, M.; Calderara, S.; Cucchiara, R. Generative adversarial models for people attribute recognition in surveillance. In Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 29 August–1 September 2017; pp. 1–6.
109. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
110. Fulgeri, F.; Fabbri, M.; Alletto, S.; Calderara, S.; Cucchiara, R. Can adversarial networks hallucinate occluded people with a plausible aspect? *Comput. Vis. Image Underst.* **2019**, *182*, 71–80. [CrossRef]
111. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
112. Papadopoulos, D.P.; Tamaazousti, Y.; Ofli, F.; Weber, I.; Torralba, A. How to make a pizza: Learning a compositional layer-based gan model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8002–8011.
113. Zhang, K.; Wu, D.; Yuan, C.; Qin, X.; Wu, H.; Zhao, X.; Zhang, L.; Du, Y.; Wang, H. Random Occlusion Recovery with Noise Channel for Person Re-identification. In Proceedings of the International Conference on Intelligent Computing. Springer, Shenzhen, China, 12–15 August 2020; pp. 183–191.
114. Tagore, N.K.; Chattopadhyay, P. A bi-network architecture for occlusion handling in Person re-identification. *Signal Image Video Process.* **2022**, *16*, 1–9. [CrossRef]
115. Wang, X.; Shrivastava, A.; Gupta, A. A-fast-rcnn: Hard positive generation via adversary for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2606–2615.

116. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
117. Han, G.; Zhou, W.; Sun, N.; Liu, J.; Li, X. Feature fusion and adversary occlusion networks for object detection. *IEEE Access* **2019**, *7*, 124854–124865. [CrossRef]
118. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef]
119. Zhao, C.; Lv, X.; Dou, S.; Zhang, S.; Wu, J.; Wang, L. Incremental generative occlusion adversarial suppression network for person ReID. *IEEE Trans. Image Process.* **2021**, *30*, 4212–4224. [CrossRef]
120. Wu, D.; Zhang, K.; Zheng, S.J.; Hao, Y.T.; Liu, F.Q.; Qin, X.; Cheng, F.; Zhao, Y.; Liu, Q.; Yuan, C.A.; et al. Random occlusion recovery for person re-identification. *J. Imaging Sci. Technol.* **2019**, *63*, 30405. [CrossRef]
121. Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; Torralba, A. Places: A 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 1452–1464. [CrossRef]
122. McCormac, J.; Handa, A.; Leutenegger, S.; Davison, A.J. Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth. *arXiv* **2016**, arXiv:1612.05079.
123. Silberman, N.; Hoiem, D.; Kohli, P.; Fergus, R. Indoor segmentation and support inference from rgbd images. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 746–760.
124. Song, S.; Yu, F.; Zeng, A.; Chang, A.X.; Savva, M.; Funkhouser, T. Semantic scene completion from a single depth image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1746–1754.
125. Armeni, I.; Sax, S.; Zamir, A.R.; Savarese, S. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv* **2017**, arXiv:1702.01105.
126. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
127. Chang, M.F.; Lambert, J.; Sangkloy, P.; Singh, J.; Bak, S.; Hartnett, A.; Wang, D.; Carr, P.; Lucey, S.; Ramanan, D.; et al. Argoverse: 3d tracking and forecasting with rich maps. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–19 June 2019; pp. 8748–8757.
128. Zhu, Y.; Tian, Y.; Metaxas, D.; Dollár, P. Semantic amodal segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1464–1472.
129. Qi, L.; Jiang, L.; Liu, S.; Shen, X.; Jia, J. Amodal instance segmentation with kins dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3014–3023.
130. Caesar, H.; Uijlings, J.; Ferrari, V. Coco-stuff: Thing and stuff classes in context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1209–1218.
131. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
132. Li, D.; Zhang, Z.; Chen, X.; Ling, H.; Huang, K. A richly annotated dataset for pedestrian attribute recognition. *arXiv* **2016**, arXiv:1603.07054.
133. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; Tian, Q. Scalable person re-identification: A benchmark. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1116–1124.
134. Li, W.; Zhao, R.; Wang, X. Human reidentification with transferred metric learning. In Proceedings of the Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Republic of Korea, 5–9 November 2012; pp. 31–44.
135. Li, W.; Zhao, R.; Xiao, T.; Wang, X. Deepreid: Deep filter pairing neural network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 152–159.
136. Zheng, Z.; Zheng, L.; Yang, Y. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3754–3762.
137. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
138. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
139. Li, Y.; Xiao, N.; Ouyang, W. Improved generative adversarial networks with reconstruction loss. *Neurocomputing* **2019**, *323*, 363–372. [CrossRef]
140. Dosovitskiy, A.; Brox, T. Generating images with perceptual similarity metrics based on deep networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 658–666.
141. Gatys, L.A.; Ecker, A.S.; Bethge, M. A neural algorithm of artistic style. *arXiv* **2015**, arXiv:1508.06576.
142. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 694–711.
143. Lim, J.H.; Ye, J.C. Geometric gan. *arXiv* **2017**, arXiv:1705.02894.

144. Li, K.; Malik, J. Amodal instance segmentation. In Proceedings of theComputer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 677–693.
145. Héder, M.; Rigó, E.; Medgyesi, D.; Lovas, R.; Tenczer, S.; Török, F.; Farkas, A.; Emődi, M.; Kadlecsik, J.; Mező, G.; et al. The past, present and future of the ELKH cloud. *Inform. Társadalom* **2022**, *22*, 128–137. [CrossRef]