A Survey on Generative Adversarial Networks for imbalance problems in computer vision tasks

Vignesh Sampath[*,1,2],Iñaki Maurtua[1], Juan José Aguilar Martín[2], Aitor Gutierrez[1]

[1] Autonomous and Intelligent Systems Unit, Tekniker,Member of Basque Research and technology Alliance, Eibar, Spain; vignesh.sampath@tekniker.es, inaki.maurtua@tekniker.es, aitor.gutierrez@tekniker.es
[2] Design and Manufacturing Engineering Department, Universidad de Zaragoza, 3 María de Luna Street,Torres Quevedo Bld,50018 Zaragoza, Spain; jaguilar@unizar.es,813406@unizar.es
[*] Corresponding author: vignesh.sampath@tekniker.es, Tel.: +34-657104544

Abstract

Any computer vision application development starts off by acquiring images and data, then preprocessing and pattern recognition steps to perform a task. When the acquired image is highly imbalanced and not adequate, the desired task may not be achievable. Unfortunately, the occurrence of imbalance problems in acquired image datasets in certain complex real-world problems such as anomaly detection, emotion recognition, medical image analysis, fraud detection, metallic surface defect detection, disaster prediction, etc., are inevitable. The performance of computer vision algorithms can significantly deteriorate when the training dataset is imbalanced. In recent years, Generative Adversarial Networks (GANs) have gained immense attention by researchers across a variety of application domains due to their capability to model complex real-world image data. It is particularly important that GANs can not only be used to generate synthetic images, but also its fascinating adversarial learning idea showed good potential in restoring balance in imbalanced datasets.

In this paper, we examine the most recent developments of GANs based techniques for addressing imbalance problems in image data. The real-world challenges and implementations of synthetic image generation based on GANs are extensively covered in this survey. Our survey first introduces various imbalance problems in computer vision tasks and its existing solutions, and then examine key concepts such as deep generative image models and GANs. After that, we propose taxonomy to summarize GANs based techniques for addressing imbalance problems in computer vision tasks into three major categories: Image level imbalances in classification, object level imbalances in object detection and pixel level imbalances in segmentation tasks. We elaborate the imbalance problems of each group, and further provide GANs based solutions in each group. Readers will understand how GANs based techniques can handle the problem of imbalances and boost performance of the computer vision algorithms.

Keywords: Generative Adversarial Networks, Imbalanced data, Computer vision, Object detection, Segmentation, Classification, Deep learning, Synthetic image, Inadequate data, Deep generative model.

1. Introduction:

Recent developments in Convolutional Neural Networks (ConvNets) have led to substantial progress in the performance of computer vision tasks applied across various domains such as self-driving cars [1], medical imaging [2], agriculture [3][4], manufacturing [5], etc. The availability of big data [6], together with increased computing capabilities is the predominant reason for the recent success. Image acquisition is the first step in the development of computer vision algorithms to acquire image data. When the acquired image is not adequate, the desired task may not be possible to achieve. Image classification, object detection and segmentation are the fundamental building blocks of the computer vision tasks. All these methods use deep ConvNets with enormous layers and have very high number of parameters that need to be tuned. Therefore, they demand a huge amount of representative data to improve their performance and generalization ability. While the amount of visual data in the world is increasing

exponentially, many of the real-world datasets suffer from several forms of imbalance. Handling imbalances in the image dataset is one of the pervasive challenges in the field of computer vision.

Image classification is the task of classifying an input image according to a set of possible classes. Classification algorithms learn to isolate important distinguishing information about an object in an image like shape or color and ignore irrelevant parts of an image such as plane background or noise. Several popular image classification architectures such as LeNet [7], AlexNet [8], VGG-16 [9], GoogLeNet [10], ResNet [11], Inception-V3 [12], DenseNet [13] take an input image then pass it through several convolutional and pooling layers. Convolutional layer helps to extract features from the input image, while a pooling layer reduces the dimension. Several successive convolutional and pooling layers may follow, depending on the layout and intent of the architecture. The result is a set of feature maps reduced in size from the original image that through a training process have learned to distill information about the content in the original image. All extracted feature maps are then transformed into a single vector that can be fed into a series of fully connected neural network to obtain a probability distribution of class scores. The predicted class for the input image can be extracted from this probability distribution.

These architectures are typically designed to work well with balanced datasets, but a common issue with real-world datasets is the imbalance of observed classes. The most commonly known imbalance problem in image classification task is the class imbalance. Class imbalance in the real-world image datasets is ubiquitous and can have an adverse effect on performance of ConvNets [14]. These datasets usually fall into four categories in terms of its size and imbalance [15]:

1. An ideal dataset is the one that contains an adequate and equal or almost equal number of samples within each class. An equal probability is assigned to all classes during training to update parameters of the network and approach the minimum value of the error function. A wide range of standard machine learning algorithms can be applied for an ideal dataset.
2. A dataset with adequate number of samples where some instances of classes are rarer than other instances of classes is said to be uneven dataset. Even though these datasets have adequate number of samples, it is costly and may not be possible for experts to manually inspect huge unlabeled datasets to annotate.
3. Tiny datasets are not easily available, and they can be difficult to collect. Such datasets have equal number of samples within each class, but they are almost impossible to collect due to privacy restriction and other reasons.
4. Absolute rare datasets have a limited number of samples and substantial class imbalance. Reasons for class imbalance in these datasets can vary but commonly the problem arises because of: (a) Very limited number of experts available for data collection; for an example, generation of medical imaging datasets requires specialized equipment and well trained medical practitioners for data acquisition (b) Enormous manual effort required to label datasets; and (c) Scarcity of samples of specific class leading to class imbalance. Consequently, the size of dataset and class imbalance problem becomes a bottleneck that prevents us from tapping the true potential of ConvNets. Figure 1 illustrates different types of datasets in terms of its size and imbalance.

Class imbalance in a dataset can stem from either between classes (inter class imbalance) or within class (intra class imbalance). Inter class imbalance occurs when a minority class contains smaller number of instances when compared to instances belonging to majority class. Classifier built using inter class imbalanced datasets are most likely to predict minority class as rare occurrences, even sometimes assumed as outlier or noise which results in misclassification of minority classes [16]. Minority classes are often of greater interest and significant, that needs to be cautiously handled. For example, in a rare disease medical diagnosis where there is a vital need to distinguish such a rare medical condition among the normal populations. Any kind of diagnosis errors will cause stress to the patient and further complications. It is therefore very important that deep learning models built using such datasets should be able to achieve a higher detection rate on minority classes.

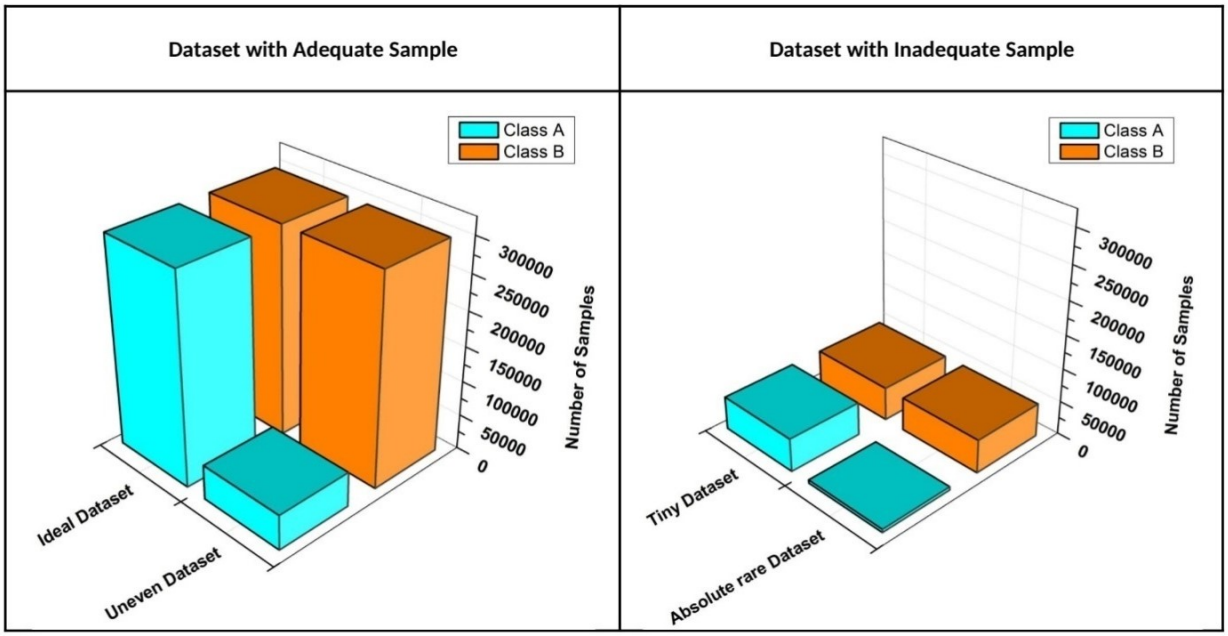

Figure 1: Distribution of different type of datasets (a) Dataset with adequate sample (b) Dataset with inadequate sample

Intra class imbalance in a dataset can also deteriorate the performance of the classifier. An Intra-class imbalance can be viewed as the attribute bias within a class, in other words inter-class imbalance in fine-grained visual categorization. For example, a class of dog sample can be further categorized by dog color, pose variations and dog breeds. Imbalances in such categories (intra class imbalance) is an unavoidable problem in datasets of many classification tasks such as modality based medical image classification [17], fine grained attribute classification [18], person re-identification [19], age [20] and pose invariant face recognition [21].

Several attempts have been made to overcome the problem of class imbalance by using different approaches and techniques. These techniques can be grouped into data-level approaches, algorithm level methods and hybrid techniques. While data level approaches modify the distribution of training set to restore balance by adding or removing instances from the training dataset, algorithm level methods change objective function of the classifier to increase the importance of the minority class. Hybrid techniques combine algorithm level methods with data level approaches. Next few paragraphs will inform readers about some of the traditional techniques available to counter the class imbalance problem.

- *Resampling:* To counteract the class imbalance problem, two types of re-sampling can be applied: One is under sampling by deleting samples from the majority class and another is oversampling by duplicating samples from the minority class [22]. Re-sampling method balances the dataset but fails to provide any additional information to the training set. The other limitations of this method include: oversampling results in over fitting problem while under sampling leads to substantial loss of information [23]. The quantity of under-sampling and oversampling is generally determined using experimental methods and empirically established [24]. In order to yield additional information to the training set, synthetic oversampling methods create new samples instead of duplicates to add equilibrium to skewed distribution. The Synthetic Minority Oversampling Technique (SMOTE) [25] is a popular synthetic oversampling method that aims to generate synthetic samples based on randomly selected K-nearest neighbors. SMOTE does not take account of the distribution of data between the classes. Adaptive synthetic sampling (ADASYN) approach [26] uses a weighted distribution for different minority class according to their learning difficulties to adaptively generate synthetic data samples. Cluster based

oversampling [27] technique divide the input space into various clusters and then incorporate sampling to alter the sample size. Many traditional synthetic oversampling techniques such as SMOTE or ADASYN are only suitable for low dimensional tabular data which restricts their application in a high dimensional image data. In addition, all the aforementioned techniques generate data by either deleting or averaging existing data, and hence may fail to improve classification performance.

- *Instance Hardness Threshold based under-sampling*: Instance hardness method identify and analyze all the instances that are overlapping among the classes [28]. It uses Instance hardness measure (IHM) to calculate the likelihood of instance misclassification and removes the entire ambiguous instance to improve the classification performance. IHM is measured at the instance level using Eq.(1).

$$IHM\langle x_i, y_i\rangle = 1 - \sum_{C} p(y_i|x_i, g)p(g|d) \quad \textbf{(1)}$$

  Where g is driven by a learning algorithm h trained on dataset d with hyper parameters β, i.e., $g = h(d, \beta)$. The term $p(y_i|x_i, g)$ is the probability assigned to instance $\langle x_i\rangle$ by g. As multiple learning algorithm is used to measure likelihood of misclassification at each instance i, IHM is a weighted sum of set of learning algorithm c with the weighing term $p(g|d)$. Higher IHM for a given instance means that higher probability of misclassification. For example, outliers, mislabeled and instances at class overlap are likely to have high IHM.
- *Augmentative Oversampling:* Data augmentation is another common technique employed to synthesis more images in a rare class of the training set [29]. Augmentation such as translation, cropping, padding, rotation and horizontal flipping introduces small modifications in the image data, but not all these modifications will improve the performance of a classifier. There is no standard method that can decide whether any particular augmentation strategy can improve results until the training process is complete. As training ConvNets is a time-consuming process [30], only restricted amount of augmentation strategy is likely to be tested before model deployment. Also, the diversity that can be obtained from small modifications of the images is relatively small. In addition to balancing classes by oversampling, augmentation techniques also severs as a kind of regularization in deep neural network architecture and hence reduce the chance of over fitting. More advanced augmentation techniques such as mixing images depends on expert knowledge [31]. A complete survey of Image data augmentation for deep learning has been compiled by Shorten et al. [31].
- *Semi-supervised learning (SSL):* SSL [32] is one of the most attractive ways to improve classification performance where we have access to small number of labeled samples x along with large amount of unlabeled samples (Uneven dataset). SSL uses the combination of supervised and unsupervised learning techniques. It makes use of small labeled samples as the training set to train the model in a supervised manner, and then use the trained model to predict on the remaining unlabeled portion of the dataset. The process of labeling each sample of unlabeled data with the individual outputs predicted for them using the trained model is known as pseudo labeling. After labeling the unlabeled data through the pseudo labeling process, classification model is trained on both the actual and pseudo labeled data. Pseudo labeling is an interesting paradigm to annotate a large scale unlabeled data that potentially take many tedious hours of human labor to manually label them. However, SSL rely on assumptions about the underlying marginal distribution of input data $p(x)$, both the labeled and unlabeled samples are assumed to have the same marginal distribution. This marginal distribution $p(x)$ should contains information about the posterior distribution $p(y|x)$. A complete list of semi supervised learning is detailed in [33].

- *Cost sensitive learning*: Majority of the classification algorithms assume that misclassification costs of both minority and majority classes are the same. Cost-sensitive learning [34] pays more attention to misclassification costs of the minority class through a cost matrix.

The most straightforward and commonly used approach in ConvNets is the data driven strategy, because deep ConvNets with enormous layers have a very high number of parameters to be tuned, it is prone to over-fitting when trained on a small sized dataset. Data level approaches inflate the training data size that severs as regularization and hence reduce the chance of over fitting in deep neural network architecture. Traditional data-level techniques (Re-sampling or Augmentative oversampling) described above; however, suffer the following drawbacks particularly when used for the class imbalance problem in high-dimensional image data, which can impede model performance.

a) Synthetic instances created using traditional data level approaches may not be the true representative of the training set.
b) Synthetic data generation is achieved either by duplication or linear interpolation which does not generate new examples that are atypical and puzzle the classifier decision boundaries, and hence fail to improve overall performance.
c) In Medical images, augmentation techniques are restricted to minor alteration on an image, as they abide by strict standards. Additionally, the types of augmentation one can use vary from problem to problem. For instance, heavy augmentations such as geometric transformations, random erasing, and mixing images might damage semantic content of the medical image.
d) Applying data augmentation in absolute rare dataset may not provide the variations required to produce distinct sample to add equilibrium to skewed distribution.
e) Dealing with the inter-class imbalance in fine-grained visual categorization is challenging because it involves modifying one or more selected attributes of images while preserving the other details.
f) Most of the techniques are designed only for binary classification problem. Multi class imbalance problem is generally considered much harder than their binary equivalents for many reasons. For Instance, there can be several combinations of minority-majority classes, i.e., they may include: 1. Few minority-Many majority classes, 2. Many minority-Few majority classes, and 3. Many minority-Many majority classes.

Class imbalance in image classification tasks has been widely explored and studied. In addition to class imbalance, there are many different forms of imbalances that can impede performance of other computer vision tasks such as object detection and image segmentation. Object detection, which deals with localization and classification of multiple objects in a given image, is another challenging and significant task in computer vision. The typical way of localizing object in an image is by drawing bounding box around the object. This bounding box can be interpreted as a collection of coordinates that define the box. Nowadays, object detection algorithms fall into two broad categories: two-stage detectors and single stage detectors. On one hand, two stage detector such as Region-based Convolutional Neural Networks (R-CNN) [35], Fast R-CNN [36], Faster R-CNN [37], Mask R-CNN [38], etc. employ a Region Proposal Network (RPN) to search objects in the first stage, and then process these region of interests for object classification and bounding-box regression in the second stage. On the other hand, single stage detectors such as Singe Shot Detection (SSD) [39], You Only Look Once (YOLO) [40], etc. perform detection on a grid that avoids spending too much time on generating region proposals. Instead of locating objects perfectly, they prioritize speed and recognition. Therefore, one stage object detectors are fast and simple, whereas two stage detectors are more accurate.

Despite the recent advances, applying object detection algorithms to the real-world datasets such as in-car video, transportation surveillance images that contain objects with large variance of scales (Objects scale imbalance) remains challenging. Physical size of a same object at different distances from the camera

would appear as different size. Singh et al. [41] showed that object level scale variation greatly affects the overall performance of object detectors. Many solutions have been proposed to address the object scale imbalance. Scale aware fast R-CNN [42] use ensemble of two object detectors, one for detecting the large and medium scale objects and other for the small scale objects, and then combine them to produce final predictions. Multi-scale Image Pyramids such as SNIP [41], SNIPER [43] use an image pyramid to build multi scale feature representation. Feature Pyramid Networks (FPN) [44] combine feature hierarchies at different scales to predict objects at different scales.

Objects in the real-world datasets only occupy a small portion of the image, while the rest of the image is background. Both single and two stage algorithms approximately evaluate about 10^4 to 10^5 locations per image [45], yet just a few locations have objects. The imbalance between foreground (object) and background can also hinder performance of the object detection algorithm. Furthermore, object detection algorithm should be invariant to deformation and occluded objects. In Pedestrian detection Dataset [46], for instance, more than 70% of pedestrians are occluded in at least one frame of a video clip and about 19% of pedestrians are occluded in all frames, where the occlusions are ranked as heavy in almost half of such cases. Dollar et al. [46] highlight that the performance of pedestrian detection using standard detector declines substantially even under partial occlusion, and drastically under severe occlusion. Data augmentation based on random erasing [47] is a frequently used technique that forces detectors to pay attention the entire object in an image, rather than just a portion of it. Yet, this technique is not guaranteed to be advantageous in all the conditions. Because skewed distributions arise even within deformed and occluded objects as some of the occlusions and deformations are uncommon that they hardly occur in practical scenario [48].

Image segmentation that classifies every pixel in an image suffers from pixel level imbalances, as are other computer vision tasks. Some of the well-known image segmentation algorithms include Fully connected network [49], SegNet [50], U-Net [51],ResUNet [52] etc. Image segmentation is essential for a variety of tasks, including: Urban scene segmentation for autonomous driving [53], Industrial inspection [54] and cancer cell segmentation [55]. Datasets of all these tasks suffer from pixel level imbalance. For example, In Urban street scene dataset [56], Pixels corresponding to sky, building and road are far numerous than pixels of pedestrian and bicyclist. Because area covered by sky, building and road are more than pedestrian and bicyclist in the image. Similarly, In brain tumour image segmentation dataset [57], MRI images have more healthy brain tissue pixels than cancerous tissue pixels. The most frequently used loss function for image segmentation task is a pixel wise cross entropy loss. This loss assigns equal weights to all the pixels, evaluates the prediction for each pixel individually and then averages over all pixels. In order to mitigate this problem, much work has been done that modify the pixel wise cross entropy loss function. The standard cross entropy loss is modified in Weighted cross entropy [51], Focal loss [45], Dice Loss [58], Generalised Dice Loss [59], Tversky loss [60], Lovász-Softmax [61] and Median frequency balancing [50], so as to assign higher importance to rare pixels. Although modified loss functions are efficient for some imbalances, such functions undergo severe difficulties when it comes to highly imbalanced datasets, as seen with medical image segmentations.

In contrast to all the traditional approaches described above, Generative adversarial Neural Networks (GANs) aim to learn underlying true data distributions from the limited available images (both minority and majority class) and then use the learned distributions to generate synthetic images. This raises an interesting question on whether GANs can be used to generate synthetic images for the minority class of various imbalanced datasets. Indeed, recent developments of GANs suggest that being capable to represent complex and high dimensional data can be used as a method of intelligent oversampling. As the exact true distribution of the high dimensional image data is very hard to learn, GANs utilize the ability of neural networks to learn a function that can approximate model distribution as close to true distribution as possible. Particularly, they do not rely on prior assumptions about the data distribution and can generate synthetic images with high visual fidelity. This significant property allows GANs to be applied to any

kind of imbalance problem in computer vision tasks. GANs can not only able to generate a fake image, but also offer a way to change something about the original image. In other words, they can learn to produce any desired number of classes (such as, objects, identities, people, etc.), and across many variations(such as, viewpoints, light conditions, scale, backgrounds, and more).There are a wide variety of GANs reported in the literature, each with their own strengths to alleviate imbalance problem in computer vision tasks. For instance, AttGAN [62], IcGAN [63], ResAttr-GAN [64],etc. are a specific variant of GANs that are commonly used for facial attribute editing tasks. They learn to synthesize not only a new face image with desired attributes but also preserves attribute independent details. Recently, GANs are combined with a wide range of existing object detection and image segmentation algorithms to overcome the problem of imbalance and improve their performance.

The original GANs architecture [65] contains two differentiable functions represented by two networks, a generator G and a discriminator D. The learning procedure of GANs is to simultaneously train a discriminator D and a generator G. It follows an adversarial two-player, zero-sum game. An intuitive way of understanding GAN is with the police and the counterfeiter anecdote. The generator network is like a group of counterfeiters trying to produce fake money and make it look like genuine. The police attempt to discover counterfeiters using fake money, yet at the same time need to let every other person spend their real money. Over time, the police show signs of improvement at identifying fake cash, and the forgers improve at faking it. In the end, the counterfeiters are compelled to make ideal copies of real money. High resolution and realistic minority class images generated using learned model distribution can be used to balance the class distribution and mitigating effect of over fitting by inflating the training dataset size. GANs solve the problem of generating data when there is not enough data to begin with and they require no human supervision. GANs can provide an efficient way to fill in holes in the discrete distribution of training data. In other words, it can transform the discrete distribution of training data to continuous, providing an additional data by nonlinear interpolation between the discrete points. Bowles et al. [66]argues that GANs offers an access to unlock additional information from a dataset. In fact, Yann LeCun, the facebook vice president and chief AI scientist, referred to GANs as "*the most interesting thing that has happened to the field of machine learning in the last 10 years*".

In this survey, as opposed to other related surveys on class imbalance, that present class imbalance in tabular data, we focus on wide range of imbalance in high dimensional image data by following a systematic approach with a view to help researchers establish a detailed understanding of GAN based synthetic image generation for the imbalance problems in computer vision tasks.

The key contributions of this survey are presented as follows:

- In this survey paper, we review current research work on GAN based synthetic image generation for the imbalance problems in visual recognition tasks. We group these imbalance problems in a taxonomic tree with three main groups: Classification, Object detection and Segmentation (Figure 2).
- Also, we provide necessary material to inform research communities about the latest development and essential technical components in the field of GAN based synthetic image generation.
- Apart from analyzing different GAN architectures, our survey focuses heavily on real world applications where GAN based synthetic images are used to alleviate imbalances and fills a research gap in the use of synthetic images for the imbalance problems in visual recognition task.

The remainder of this paper is organized as follows: "Deep Generative image models" section gives readers necessary background information on generative models. "Generative adversarial Neural Network" section discusses selected GAN variants from the architecture, algorithm, and training tricks perspective in detail. In "Taxonomy of class imbalance in visual recognition tasks" section, we provide a brief explanation on various types of imbalances encountered in visual recognition tasks and how the

GAN based synthetic image is used to rebalance, followed by GAN variants from the application perspective. "Discussion and future work" section identifies and enumerates our perspective and possible future research direction. Finally, we conclude the paper "conclusion" in section.

Figure 2: Proposed taxonomy for the review of imbalanced problem in computer vision tasks

2. Deep Generative image models:

Deep Generative model is an important family of unsupervised learning methods that are dedicated to describe the underlying distribution of unlabeled training data and learn to generate brand new data from that distribution. They are capable of uncovering underlying features in the under and over-represented image datasets, then use this information to create fair and representative datasets. Image data is pixel values encoded into three-dimensional stacked array, made up of height, width, and three-color channels. Modeling the distribution of image data is extremely challenging as natural images are high dimensional and highly structured. This challenge has led to a rich variety of neural network based generative image models, each having their own advantages. Research into neural network based generative models for image generation has a long history. Restricted Boltzmann Machines [67][68][69] and their deep variants[70][71][72] are a popular class of probabilistic models for image generation. Now the generative image models can be grouped into three broad categories: 1. Autoregressive models, 2. Latent variable models and 3. Adversarial learning based models.

Autoregressive models (ARs) aim to estimate a distribution over images (density estimation) using a joint distribution of the pixels in the image by casting it as a product of conditional distributions [73]. ARs

transform the problem of joint modeling into sequence problem, where, given all the pixels previously generated, one learns to predict the next pixel. But a highly powerful sequence model is needed to model the highly non-linear and long span auto correlations between the pixels. Based on this idea, many research articles have been published that use different sequence models from deep learning to model the complex conditional distribution. Fully visible belief networks (FVBNs) [74][75]are one of the tractable explicit density models that use chain rule to factorize likelihood of an image x into product of one dimension distributions, where $n \times n$ pixels in the grey scale image is taken row by row as a one dimensional sequence $x_1, x_2, x_3 \ldots, x_{n^2}$. The joint likelihood $p(x)$ is explicitly computed as the product of the conditional probabilities over the pixels. The conditional distribution of each pixel in an image is calculated as shown in Eq. (2).

$$p(x) = \prod_{j=1}^{n^2} p(x_j | x_1, x_2, \ldots x_{j-1}) \quad \textbf{(2)}$$

Given all the preceding pixels $x_1, x_2 \ldots x_{j-1}$, the value $p(x_j | x_1, x_2, \ldots x_{j-1})$ is the probability of the j-th pixel x_j. Each pixel is dependent on previous pixels that have been already generated. The pixel generation starts from corner, continues pixel by pixel and row by row. In the case of RGB image, each pixel values in an individual RGB color is jointly computed by three values, one for each of the RGB color channels. The conditional distribution $p(x_j | X < j)$ can be rewritten as following product (Eq. (3)) where green channel is conditioned on channel red and blue channel is conditioned on channels red and green.

$$p(x_j, R | X < j) p(x_j, G | X < j, x_j, R) p(x_j, B | X < j, x_j, R, x_j, G) \quad \textbf{(3)}$$

Generating an image pixel by pixel using this approach is sequential, computationally intense, and very slow process as each of the colour channels is conditioned on the other channels as well as on all the pixels generated previously.

(a) (b)

Figure 3: Autoregressive models train a network that models conditional distribution of each pixel given all previous pixels. The image is processed pixel-by-pixel in (a) Raster scan order and (b) Sequentially predicts pixels.

Neural Autoregressive Density Estimator (NADE) [76] aims to learn a joint distribution using a neural network to parametrize the factors of $p(x)$. The output layer of the NADE is designed to predict n conditional probability distributions, each node in the output layer corresponds to one of the factors in the joint distribution. Hidden representation for each output node is computed using only relevant inputs, i.e. only previous $i - 1$ input variables are connected to the i^{th} output. By implementing Neural network, NADE allows weights sharing that reduces number of parameters to learn a joint distribution using stochastic gradient descent.

Recurrent neural networks (RNN) have been proved to excel at various sequential tasks, such as speech recognition, speech synthesis, handwriting recognition, and image to text. Particularly, Long Short-Term Memory (LSTM) layers [77] are the robust RNN architecture for modelling long range sequence data with auto correlations like time series data, natural languages etc. In order to have a long-term memory, LSTM layer adds gates to the RNN. It has an input to state component and a recurrent state to state component that together determine the gates of the layer. Theis et al.[78] used spatial LSTM (sLSTM), a

multi-dimensional LSTM which is suitable for image modelling because of its spatial structure. However, an immense amount of time is needed to train the LSTM layers considering number of pixels in the larger datasets such as CIFAR-10 [79] and ImageNet [80].

Van den Oord et al. [81] designed two variants of recurrent image model: PixelRNN and PixelCNN. The pixel distributions of the natural images are modeled with two-dimensional LSTM (spatial LSTMs) and convolutional networks in PixelRNN and PixelCNN respectively. Convolution operation enables PixelCNNs to generate pixels faster than PixelRNNs, given the large number pixels in natural images. But typically, PixelRNNs achieve higher performance when compared to PixelCNNs. Gated PixelCNN [82] is another interesting paradigm to generate diverse natural images with density model conditioned on prior information along with previously generated pixels. The prior information h in Eq.(4) can be any vector, including class labels or tags.

$$p(x|h) = \prod_{j=1}^{n^2} p(x_j|x_1, x_2, \ldots x_{j-1}, h) \quad \textbf{(4)}$$

A lot of work on improving performance of PixelCNN has been reported in literature by introducing new architectures, loss functions and different training tricks. PixelCNN++ [83] enhances performance of PixelCNN by proposing numerous modifications while retaining its computational performance. Major modifications include: 1. Intensity of a pixel is viewed as 8-bit discrete random variables and modeled using 256-softmax output in pixelCNN. In contrast, PixelCNN++ uses discretized logistic mixture likelihood to model each pixel as real valued output. 2. It simplifies the model structure by conditioning on entire pixels, instead of RGB sub space. 3. PixelCNN++ employs down-sampling by using convolution of stride 2 in order to capture structure at multiple resolutions 4.Short cut connections are added to compensate the loss of information due to down-sampling. 5. PixelCNN++ also introduces model regularization using dropouts. Pixel Snail [84] incorporates self-attention mechanism in PixelCNN to have access to long term temporal information.

Latent variable models, on the other hand, aim to represent high dimensional image data (observable variables) into lower dimensional latent space (latent variables). Latent variables as opposed to observable variables are variables that are not directly observed but inferred through a model from other variables that are observed directly. One advantage of using latent variable is that it reduces dimensionality of data. High dimensional observable variables can be aggregated in a model to represent an underlying concept making it easier to understand the data.

Autoencoders are one of the latent variable models that takes unlabeled high dimensional image data x, and after encoding them lower dimensional feature representation z, tries to reconstruct them as accurately as possible. The lower dimensional feature z is a compressed representation of an input image, as a result, the autoencoder must decide which of the features in an image are the most important, essentially acting as feature extraction engine or dimensionality reduction. They are typically very shallow neural network, and are usually comprised of an input layer, an output layer, and a hidden layer. Autoencoders with nonlinear encoder and decoder functions learn to project image data onto a nonlinear manifold, powerful nonlinear generalization compared to principle component analysis (PCA). They are trained with back propagation, using a metric called Reconstruction loss. Reconstruction loss measures the amount of information that was lost when an autoencoder tried to reconstruct the input, using pixel wise L1 or L2 distance. In other words, pixel wise distance between original images x and reconstructed images $\hat{x}$. Autoencoders with a small loss value can produce reconstructed image that look very similar to the original image.

Traditionally, autoencoders are used for data denoising, data compression and dimensionality reduction. There are many variants of autoencoder proposed in the literature [85], [86],[87],[88],[89],[90].Deep

autoencoders [86] uses a stack of layers as encoder and decoder instead of limiting to a single layer. Sparse autoencoders [87] have a larger number of hidden neurons than the input or output neurons, but only a fraction of hidden neurons are permitted to be active at once. ConvNets are used as encoder and decoder in convolutional autoencoders [91]. In order to learn a function that is robust to minor variations in its training dataset, contractive autoencoders [89] add a penalty term to its objective function. Denoising autoencoders [85] are stochastic form of the basic autoencoder that add white noise to the training data to reduce a situation of learning the identity function.

An autoencoder is tweaked to predict the n-conditional distributions rather than just reconstructing the inputs in Masked Autoencoder Density Estimator (MADE) [92]. In the standard fully connected autoencoder i^{th}output unit depends on all the input units, but in order to predict the conditional distributions, i^{th} output unit should depend only on previous $i-1$ input variables. MADE modifies the autoencoder using binary mask matrix to ensure each output unit is connected only to relevant input units. As opposed to autoencoders that are used for an image abstraction, MADE is designed for image generation using learnt distribution.

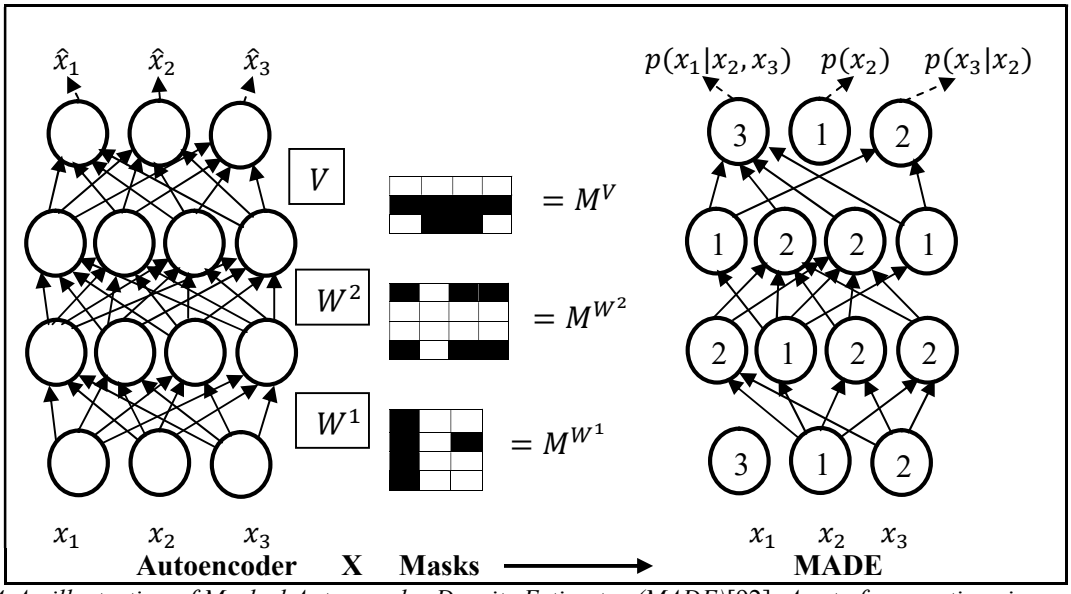

Figure 4:An illustration of Masked Autoencoder Density Estimator (MADE)[92]. *A set of connections in an autoencoder is removed using multiplicative binary masks, such that each output unit is connected only to relevant input units.*

Variational Auto Encoders (VAEs) [90] are the most popular class of autoencoders. In VAEs, the encoder instead of outputting a latent vector directly, outputs mean μ and variance σ vectors which constitutes latent probability distributions $q_\emptyset(z|x)$ from which a latent vector is sampled. This means that given the same input image, no two latent vectors sampled are the same which forces the decoder to learn the mapping from a region of a latent space to a reconstruction rather than just from a single point resulting in much smoother reconstructed image. Unlike traditional autoencoders, which are only be able to reconstruct image similar to training set, VAEs can generate new image close to training set. VAEs are trained by maximizing the variational lower bound (Eq.(5)) also known as evidence lower bound.

$$\mathcal{L}_{VAE}(\theta,\emptyset;x,z) = \underbrace{D_{KL}(q_\emptyset(z|x)||p(z))}_{Latent\ loss} - \underbrace{E_{z\sim q_\emptyset(z|x)}\big(logP_\theta(x|z)\big)}_{Reconstruction\ loss} \quad \textbf{(5)}$$

The first term in Eq.(5) is the Latent loss which regularizes the distribution of q to be Gaussian normal distribution $\mathcal{N}(0,1)$ by minimizing Kullback-Leibler divergence (KL divergence). KL divergence measures similarity between the latent probability distribution and the prior distribution using relative entropy. KL divergence from probability distribution q to p is defined to be

$$D_{KL}(q||p) = \sum_{x} q(x) log \frac{q(x)}{p(x)} \quad \textbf{(6)}$$

The latent loss is high when the latent probability distribution does not resemble a standard multivariate Gaussian and it is low when the resemblance between those two distributions is close. Given input data x, a probabilistic encoder encodes them to latent representation z with distribution $q_{\emptyset}(z|x)$ and a probabilistic decoder decodes $p_{\theta}(x|z)$. Latent loss enforces the posterior distribution of latent representation z to match with an arbitrary prior distribution $p(z)$. In other words, it imposes a restriction in z, such that input data x are distributed in a latent space following a specified arbitrary prior distribution. The second term, reconstruction loss is pixel wise Binary cross entropy between original image x and reconstructed image $\hat{x}$.

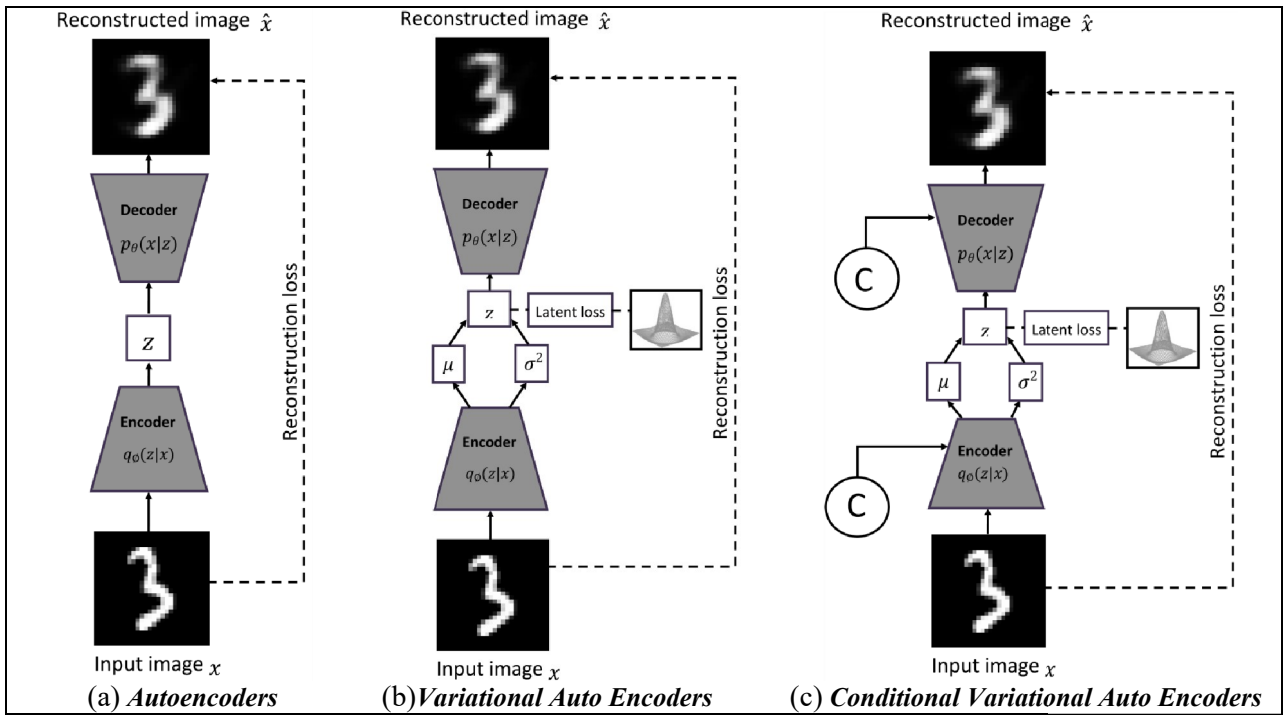

Figure 5: The architecture of (a) Autoencoders; (b) Variational Auto Encoders; (c) Conditional Variational Auto Encoders

A numerous modifications have been made over basic VAEs that was initially introduced in [90]. The Conditional VAE (CVAE) [93] is a conditioned version of standard VAEs (Figure 5C) to generate diverse reconstructed images conditioned on additional information such as class labels, facial attributes etc. Variational lower bound of CVAE is written as

$$\mathcal{L}_{CVAE}(\theta, \emptyset; x, z, c) = D_{KL}(q_{\emptyset}(z|x,c)||p(z,c)) - E_{z\sim q_{\emptyset}(z|x)}\big(logP_{\theta}(x|z,c)\big) \quad \textbf{(7)}$$

Beta VAE (β-VAE) [94] is another modified form of original VAE intended to learn disentangled latent representations that capture the independent features of a given image. It introduces additional hyper parameter β that balances the latent and reconstruction loss. Variational lower bound of β-VAE is defined as

$$\mathcal{L}_{\beta-VAE}(\theta, \emptyset, \beta; x, z) = \beta[D_{KL}(q_{\emptyset}(z|x)||p(z))] - E_{z\sim q_{\emptyset}(z|x)}\big(logP_{\theta}(x|z)\big) \quad \textbf{(8)}$$

When $\beta = 1$ in Eq.(8), it corresponds to the standard VAE framework. β-VAE with $\beta > 1$ pushes the model to learn disentangled representation. Deep Convolutional Inverse Graphics Network (DC-IGN) [95] replaced feed forward neural networks in the encoder and decoder of VAEs with convolution and de-convolution operators respectively. Importance weighted VAE (IWVAE) [96] learns richer and more complex latent space representation than VAEs from importance weighting. Convolutional VAE is combined with the PixelCNN in PixelVAE[97] and Variational lossy autoencoder [98]. Deep Recurrent Attentive Writer (DRAW) [99] networks combine spatial attention mechanism with a sequential variational autoencoder. In order to avoid problems of posterior collapse, Vector Quantized VAE (VQ-VAEs) [100] learns discrete latent representation instead of continuous normal distribution. VQ-VAEs combine VAEs with ideas from vector quantization to get a sequence of discrete latent variables. VQ-VAE 2 [101] is a Hierarchical multi-scale VQ-VAE combined with self-attention mechanism for generating high resolution images.

Adversarial models try to model the distribution of the real data through an adversarial process. Generative adversarial neural networks based on game theory, introduced by Goodfellow et.al [65] in 2014, is arguably one the best innovation in recent years. The word adversarial in generative adversarial networks means that the two networks the generator and the discriminator are in a competition with each other. The learning procedure of GAN is to simultaneously train a discriminator D and a generator G. The generator network takes a noise vector z in a latent space as input, then runs that noise vector through a differentiable function to transform the noise vector z to create fake but plausible image $x : G(z) \rightarrow x$. At the same time, the discriminator network, which is essentially a binary classifier, tries to distinguish between the real images (label 1) and artificially generated images by generator network (label 0): $D(x) \rightarrow [0,1]$. Therefore, the objective function of GANs can be defined as

$$\min_{G}\max_{D} V(D,G) = E_{x\sim p_r(x)}[logD(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))] \quad \textbf{(9)}$$

Given random noise vector z and real image x , the generator attempts to minimize $\log(1 - D(G(z))$ and the discriminator attempts to maximize $logD(x)$ in Eq.(9). For fixed G, the optimal D is given by

$$D^*(x) = \frac{p_r(x)}{p_g(x) + p_r(x)} \quad \textbf{(10)}$$

Theoretically, when G is trained to its optimal, the generated data distribution $p_g(x)$ gets closer to the real data distribution $p_r(x)$. If $p_g(x) = p_r(x)$, $D^*(x)$ in Eq.(10) becomes ½. This means that discriminator is maximally puzzled and cannot distinguish fake images from real ones. When the discriminator D is optimal, the loss function for the generator Gcan be visualized by substituting in $D^*(x)$ Eq.(9).

$$G^* = \max_{D} V(G, D^*) = E_{x\sim p_r(x)}[logD^*(x)] + E_{x\sim p_g(x)}[\log(1 - D^*(x))]$$
$$= E_{x\sim p_r(x)}\left[log\frac{p_r(x)}{\frac{1}{2}[p_g(x)+p_r(x)]}\right] + E_{x\sim p_g(x)}\left[log\frac{p_g(x)}{\frac{1}{2}[p_g(x)+p_r(x)]}\right] - 2log2 \quad \textbf{(11)}$$

The definition of Jensen-Shannon divergence (D_{JS}) between two probability distributions $p_g(x)$ and $p_r(x)$ is defined as

$$D_{JS}(p_r||p_g) = \frac{1}{2}D_{KL}(p_r||\frac{p_r + p_g}{2}) + \frac{1}{2}D_{KL}(p_g||\frac{p_r + p_g}{2}) \quad \textbf{(12)}$$

Therefore, Eq.(11) is equal to

$$G^* = 2D_{JS}(p_r(x)||p_g(x)) - 2log2 \quad \textbf{(13)}$$

Essentially, the loss for the generator G minimizes the Jensen-Shannon divergence between the generated data distribution $\mathrm{p_g(x)}$ and the real data distribution $\mathrm{p_r(x)}$ when discriminator D is optimal. Jensen-Shannon divergence is a smooth, symmetric version of the KL divergence. Huszar [102] believes that the main reason behind the great success of GANs is replacing asymmetric KL divergence loss function in classical approach to symmetric JS divergence.

Mean squared error used in latent variable models such as autoencoder, averages all the possible features in an image and generate blurry images. In contrast, adversarial loss preserves the features using discriminator networks that detect an absence of any features as an unrealistic image. An example of this is the study carried out by Lotter et al. [103], in which models trained using mean square loss and adversarial loss to predict the next image frame in a video sequence are compared. A model trained using mean square loss generates blurry image as shown in Figure 6, where ear and eyes are not sharply defined as they could be. Using an additional adversarial loss, features like the eyes and ear remain preserved very well, because an ear is the recognizable pattern and the discriminator network would not accept any sample that is missing an ear.

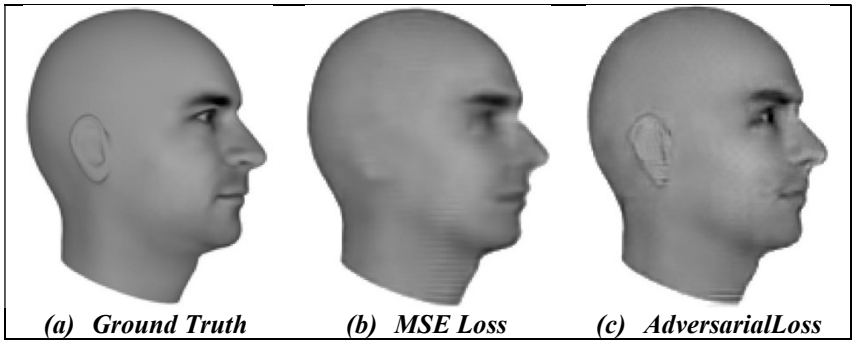

Figure 6: An illustration of the importance of an adversarial loss [103].

This section has attempted to provide readers a brief introduction to current state of deep generative image models. A quick summary of this section is depicted below in Figure 7.

Despite remarkable achievements in generating sharp and realistic images, GANs suffer from certain drawbacks.

- ***Non convergence:*** Both generator and discriminator networks in GANs are trained simultaneously using gradient descent in a zero-sum game. As a result, improvement of generator network comes at the expense of discriminator and vice versa. Hence there is no guarantee of GANs convergence.
- ***Mode collapse****:* Generator network achieves a state where it continues to generate sample with little variety, although trained on diverse datasets. This form of failure is referred as mode collapse.
- ***Vanishing gradient problems:*** If the discriminator is perfectly trained early in the training process, then there would be no gradients left to train generator due to vanishing gradients.

Therefore, many GAN-variants have been proposed to overcome these drawbacks. These GAN-variants can be grouped into three categories:

(1) *Architecture Variants*. In terms of architecture of generator and discriminator networks, the first proposed GANs use the Multi- layer perceptron (MLP). Owing to the fact that ConvNets work

well with high resolution image data taking into account of the spatial structure of data, a Deep Convolutional GAN (DCGAN) [104] replaced the MLP with the deconvolutional and convolutional layers in generator and discriminator networks respectively.

Autoencoder based GANs such as AAE [105], BiGAN [106], VAE-GAN [107], DEGAN [108],VEEGAN [109] etc., have been proposed to combine the reconstruction power of autoencoders with the sampling power of GANs.

Conditional based GANs like Conditional GAN (CGAN) [110], Auxiliary Classifier GAN (ACGAN) [111], VACGAN [112], infoGAN [113],SCGAN [114] focused on controlling mode of data being generated by conditioning model on conditional variable.

(2) *Training tricks*. GANs are difficult to train. Improved trainings tricks such as feature matching, minibatch discrimination, historical averaging, one-sided label smoothing and Two Time-Scale Update Rule have been suggested to ensure that GANs converge to achieve Nash equilibrium.

(3) *Objective Variants*. In order to improve the stability and overcome vanishing gradient problem, different objective functions have been explored in [115], [116], [117], [118], [119], [120], [121], [122].

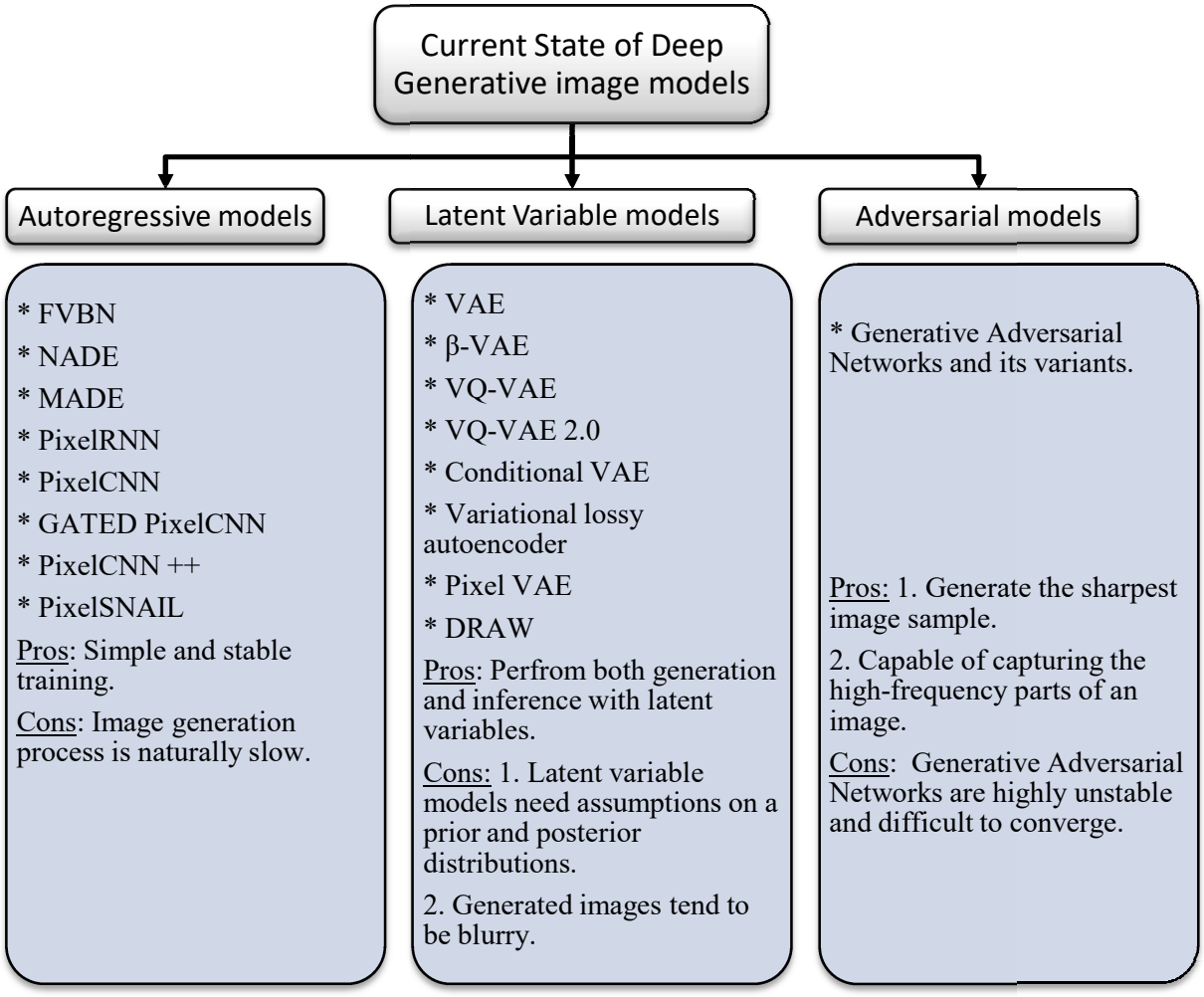

Figure 7: Comparative summary of Deep generative models discussed in section 2.

The following section of this review moves on to describe in greater detail the selected GAN variants.

3. Generative Adversarial Networks:

3.1. *Architecture variants:*

The performance and training stability of GANs are highly influenced by architecture of the generator and the discriminator networks. Various architecture variants of GANs have been proposed that adopt several techniques to improve performance and stability.

i. Conditional based GAN Variants

The standard GAN [65] architecture does not have any control on modes of the data being generated. Van den Oord et al. [82] argue that the class conditioned image generation can significantly enhance the quality of generated images. Several conditional based GANs have been proposed that learn to sample from a conditional distribution $p(x|y)$ instead of marginal $p(x)$. Conditional based GANs variants can be classified into two groups: 1. Supervised and 2. Unsupervised conditional GANs.

Supervised conditional GANs variants require a pair of image and corresponding prior information such as class label. The prior information could be class labels, textual descriptions, or data from other modalities.

cGAN:Mirza and Osindero [110] proposed conditional Generative Adversarial Network (cGAN), to have a control on kind of data being generated by conditioning the model on prior information y. Both discriminator and generator in cGAN are conditioned by feeding y as additional input. Using this prior information, cGAN is guided to generate output images with desired properties during generation process.

ACGAN: Auxiliary classifier Generative Adversarial Network (ACGAN) [111] is an extension of the cGAN architecture. The discriminator in the ACGAN receives only the image, unlike the cGAN that gets both the image and class label as input. It is modified to distinguish real and fake data as well as reconstruct class labels. Therefore, in addition to real fake discrimination, the discriminator also predicts class label of the image using auxiliary decoder network.

VACGAN: The major problem with ACGAN is that it will affect the training convergence because of mixing the loss of classifier and discriminator into a single loss. Versatile Auxiliary Generative Adversarial Network (VACGAN) [112] separates out classifier loss by introducing classifier network in parallel to the discriminator.

No prior information is used in unsupervised conditional GAN variants to control on modes of the image being generated. Instead, feature information such as hair color, age, gender etc. is learned during training process. Therefore, they need an additional algorithm to decompose the latent space into disentangled latent vector$c$, which contains the meaning features, and standard input noise vector z. The content and representation of an image is then controlled by noise vector z and disentangled latent vector$c$ respectively.

Info-GAN: Information maximizing Generative Adversarial Network (Info-GAN) [113] splits an input latent space into the standard noise vector z and additional latent vector$c$. The latent vector c is then made meaningful disentangled representation by maximizing the mutual information between latent vector c and generated images $G(z, c)$ using additional Q network.

SC-GAN: Similarity constraint Generative Adversarial Network (SC-GAN) [114] attempts to learn disentangled latent representation by adding the similarity constraint between latent vector c and generated images $G(z, c)$. Info-GAN uses an extra network to learn disentangle representation, while SC-

GAN only adds an additional constraint to a standard GAN. Therefore, SCGAN simplifies the architecture of Info-GAN.

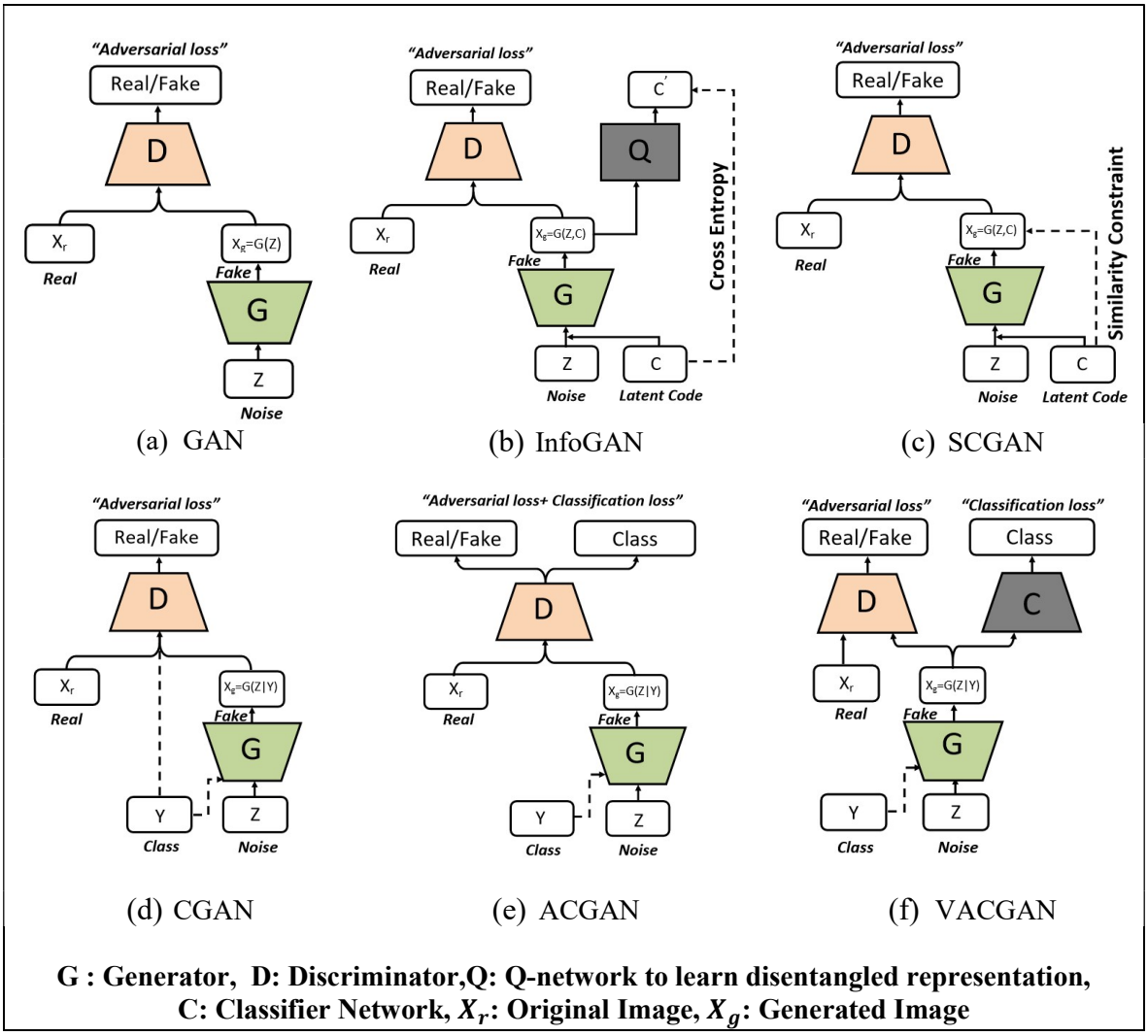

Figure 8: A schematic view of (a) the vanilla GAN and (b)-(f) variants of Conditional GANs

ii. Convolutional based GAN:

DCGAN: Deep Convolutional Generative Adversarial Network (DCGAN) [104] is the first work that deploys convolutional and transpose-convolutional layers in the discriminator and generator, respectively. The salient features of the DCGAN architecture are enumerated as follows:

- First, the generator in DCGAN composed of fractional convolutional layers, batch normalization layers and ReLU activation functions.
- Second, the discriminator is comprised of strided convolutional layers, batch normalization layers and Leaky ReLU activation functions.
- Third, uses Adaptive Moment Estimation (ADAM) optimizer instead of stochastic gradient descent with momentum.

iii. Multiple GANs

In order to accomplish more than one goal, several frameworks extend the standard GAN to either multiple discriminators, generators, or both.

ProGAN: In an attempt to synthesize higher resolution images Progressive Growing of Generative Adversarial Network (ProGAN) [123] stacks each layer of the generator and discriminator in a progressive manner as training progresses.

LAPGAN: Laplacian Generative Adversarial Network (LAPGAN) [124] is proposed for the generation of high quality images. This architecture uses cascade of ConvNets within a Laplacian pyramid framework. LAPGAN utilizes several Generator-Discriminator networks at multiple levels of a Laplacian Pyramid for an image detail enhancement. Motivated by the success of sequential generation, Im et.al[125] introduced Generative Recurrent Adversarial Networks (GRAN) based on recurrent network that generate high quality image in a sequential process, rather than in one shot.

D2GAN: Dual discriminator Generative Adversarial Network (D2GAN) [126] employs two discriminators and one generator to address the problem of mode collapse. Unlike GANs, D2GAN formulates a three-player game that utilizes two discriminators to minimize the KL and reverse KL divergences between true data and the generated data distribution.

MADGAN: Multi-agent diverse Generative Adversarial Network (MADGAN) [127] incorporates multiple generators that discover diverse modes of the data while maintaining high quality of generated images. To ensure that different generators learn to generate images from different modes of the data, the objective of discriminator is modified to detect the generator which generated the given fake image along with discriminating the real and fake images.

CoGAN: Coupled GAN(CoGAN) [128] is used for generating pair of like images in two different domain. CoGAN is composed of set of GANs – GAN1 and GAN2, each accountable for synthesizing images in one domain. It leans a joint distribution from two-domain images which are drawn individually from the marginal distributions.

CycleGAN and DiscoGAN [129] use two generators and two discriminators to accomplish unpaired image to image translation task. CycleGAN [130] adopts the concept of cycle consistency from machine translation, where a sentence translated from English to Spanish and translate it back from Spanish to English should be identical.

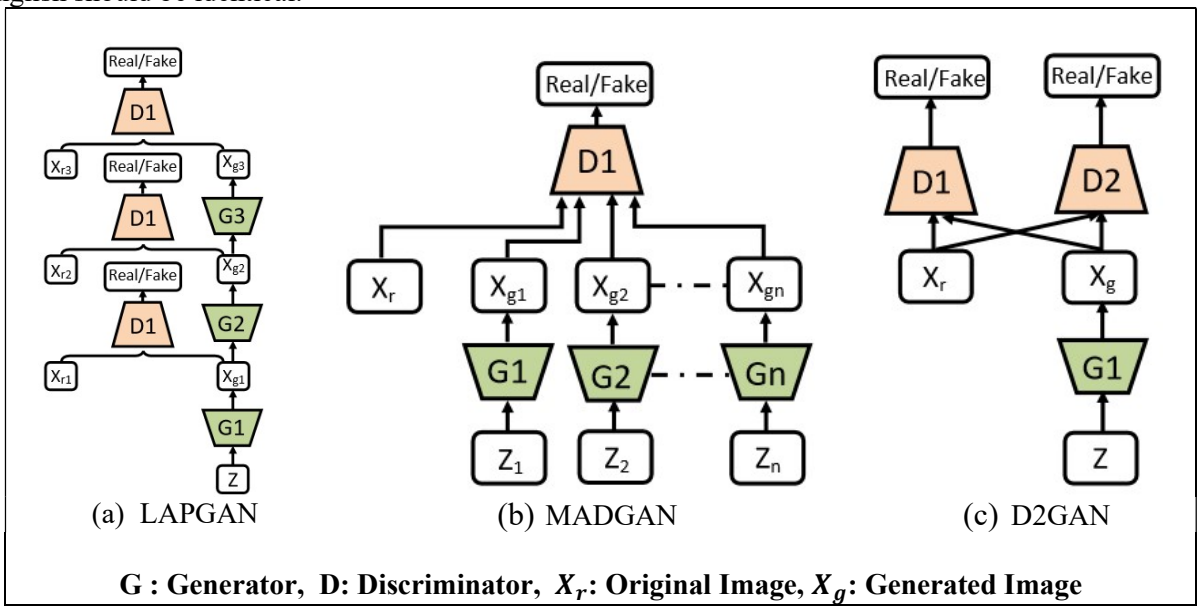

Figure 9: A schematic view ofVariants of GANs with multiple discriminators and generators: (a) LAPGAN, (b) MADGAN and (c) D2GAN

iv. Autoencoder based GAN Variants

The standard GANs architecture is unidirectional and can only map from latent space z to data space x, while autoencoders are bidirectional. The latent space learned by encoders is the distribution that contains compressed representation of the real images. Several variants of GANs that combine GAN and encoders architecture are proposed to make use of the distribution learned by encoders. Attributes editing of an image directly on data space x is complex as image distributions are highly structured and high dimensional. Interpolation on latent space can facilitates to render complicated adjustments in the data space x.

DEGAN: In standard GANs architecture, the input to the generator network is the noise vector that are randomly sampled from a Gaussian distribution $N(0,1)$, which may create a deviation from the true distribution of real images. Decoder Encoder Generative adversarial Network (DEGAN) [108] adopt decoder and encoder structure from VAE, pretrained on the real images. The pretrained decoder and encoder structure transform random Gaussian noise to distribution that contains intrinsic information of the images which is used as input of the generator network.

VAEGAN: Variational autoencoder Generative Adversarial Network (VAEGAN) [107] jointly train a VAE and GAN by replacing the decoder of VAE with GAN framework. VAEGAN employs feature wise adversarial loss of GAN in lieu of element wise reconstruction loss of VAE to improve quality of image generated by VAE. In addition to latent loss and adversarial loss, VAEGAN uses content loss also known as perceptual loss that compares two images based on high level feature representation from pre-trained VGG Network [9].

AAE: Unlike VAEGAN that discriminates in data space, adversarial autoencoders (AAE) [105] imposes a discriminator on the latent space as learning the latent code distribution is simpler than data distribution. The discriminator network discriminates between a sample drawn from latent space and from the distribution $p(z)$ that we are trying to model.

ALI and BiGAN: In addition to generator network, Adversarially Learned Inference (ALI) [106] model and Bidirectional Generative Adversarial Network (BiGAN) contain an encoder component E that simultaneously learn inverse mapping of the input data x to the latent code$z$. Unlike other variants of GAN where the discriminator network receives only real or artificially generated images, in the BiGAN and ALI model, the discriminator network receives both image and latent code pair.

VEEGAN [109]**:** addresses the problem of mode collapse through addition of reconstruction network that reverse the action of generator network. Reconstruction network takes in synthetic images then transform them to noise, while generator network takes noise as an input and reconstruct them into synthetic image. In addition to adversarial loss, difference between the reconstructed noise and initial noise is used to train the network. Both generator and reconstruction networks are jointly trained, which encourages generator network to learn true distribution, hence solve the mode collapse problem.

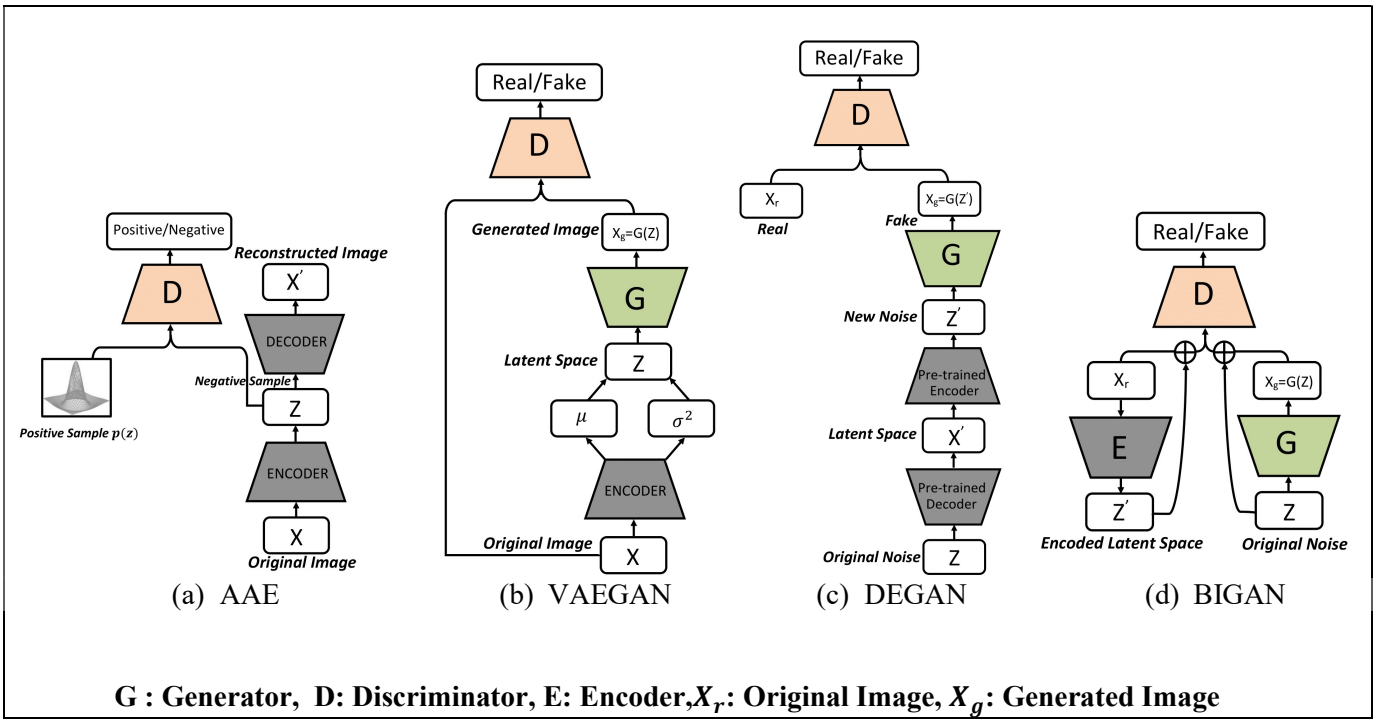

Figure 10: A schematic view of Variants of GANs based on Encoder and decoder architecture: (a) AAE, (b) VAEGAN, (c) DEGAN and (d) BIGAN

Several other GANs have been proposed for image super resolution. The goal of super resolution is to upsample low resolution image to a high resolution one. Ledig et al. proposed Super-Resolution GAN (SRGAN) [131] for image super resolution,which takes poor quality image as input, and generates high quality image with 4x resolution. The generator of the SRGAN uses very deep convolutional layers with residual blocks. In addition to an adverasarial loss, SRGAN includes a content loss. The content loss is computed as the euclidean distance between the feature maps of the generated high quality image and the ground truth image, where feature maps are obtained from a pretrained VGG19 [132] network. Zhang et al.[133] combined self attention mechanism with GANs (SAGAN) to handle long range dependencies that make generated image look more globally coherent. Image-to-image translation GANs such as Pix2Pix GAN [134], Pix2pix HD GAN [135], and CycleGAN [129] learn to map an input image from a source domain to an output image from a target domain.

Categories	*GAN Type*	*Main Architectural Contributions to GAN*
Basic GAN	GAN [65]	Use Multi-layer perceptron's in the generator and discriminator
Convolutional Based GAN	DCGAN [104]	Employ Convolutional and transpose-convolutional layers in the discriminator and generator respectively
	PROGAN [123]	Progressively grow layers of GAN as training progresses
Condition based GANs	cGAN [110]	Control kind of image being generated using prior information
	ACGAN [111]	Add a classifier loss in addition to adversarial loss to reconstruct class labels
	VACGAN [112]	Separate out classifier loss of ACGAN by introducing separate classifier network parallel to the discriminator
	infoGAN [113]	Learn disentangled latent representation by maximizing mutual information between latent vector and generated images
	SCGAN [114]	Learn disentangled latent representation by adding the similarity constraint on the generator

Latent representation based GANs	DEGAN [108]	Utilize the pretrained decoder and encoder structure from VAE to transform random Gaussian noise to distribution that contains intrinsic information of the real images
	VAEGAN [107]	Combine VAE and GAN
	AAE [105]	Impose discriminator on the latent space of the autoencoder architecture
	VEEGAN [109]	Add reconstruction network that reverse the action of generator network to address the problem of mode collapse
	BiGAN [106]	Attach encoder component to learn inverse mapping of data space to latent space
Stack of GANs	LAPGAN [124]	Introduce Laplacian pyramid framework for an image detail enhancement
	MADGAN [127]	Use multiple generators to discover diverse modes of the data distribution
	D2GAN [126]	Employ two discriminators to address the problem of mode collapse
	CycleGAN [129]	Use two generators and two discriminators to accomplish unpaired image to image translation task
	CoGAN [128]	Use two GANs to lean a joint distribution from two-domain images
Other Variants	SAGAN [133]	Incorporate self-attention mechanism to model long range dependencies
	GRAN [125]	Recurrent generative model trained using adversarial process
	SRGAN [131]	Use very deep convolutional layers with residual blocks for image super resolution

Table 1: An overview of GANs variants discussed in Section 3.1.

3.2. *Objective variants:*

The main objective of GAN is to approximate the real data distribution. Hence, minimizing distance between the real data distribution (p_r) and the GAN generated data distribution (p_g) is a vital part of training GAN. As stated in section 2, standard GAN [65] uses Jensen Shannon divergence to measure similarity between real and generated data distributions $D_{JS}(p_r||p_g)$. However, JS divergence fails to measure distance between two distributions with negligible or no overlap. To improve performance and achieve stable training of GAN, several distances or divergence measures have been proposed instead of JS divergence.

WGAN: Wasserstein Generative Adversarial Network (WGAN) [115] replaces JSD from the standard GAN with the Earth mover Distance (EMD). EMD also known as Wasserstein Distance (WD) can be interpreted informally as minimum amount of work to move earth (quantity of mass) from the shape of one distribution p(x) to that of another distribution q(x) so as to match shape of both the distributions. WD is smooth and can provide meaningful distance measure between distributions with negligible or no overlap. WGAN imposes an additional Lipchitz constraint to use WD as the loss in the discriminator, where it deploys weight clipping to enforce weights of the discriminator to satisfy Lipchitz constraint after each training batch.

WGAN-GP: Weight clipping in discriminator of a WGAN greatly diminishes its capacity to learn and often fail to converge. WGAN-GP [116] is an extension of WGAN that replaces weight clipping with gradient penalty to enforce discriminator to satisfy Lipchitz constraint. Furthermore, Petzka et al. [117] proposed a new regularization method, also known as WGAN-LP, that enforce the Lipschitz constraint.

LSGAN: Least squares Generative Adversarial Network (LSGAN) [118] deploys least square loss instead of the cross entropy loss in discriminator of the standard GAN to overcome the problem of Vanishing gradient as well as improving quality of generated image.

EBGAN: Energy Based GAN (EBGAN) [119] uses auto-encoder architecture to construct the discriminator as an energy function instead of a classifier. The Energy of EBGAN is the mean squared reconstruction error of an autoencoder, providing lower energy to the real images and high energy to generated images. EBGAN exhibits fast and more stable behavior than standard GAN during training.
Same as EBGAN, Boundary Equilibrium GAN (BEGAN) [120], Margin adaptation GAN [121] and dual agent GAN [122] also deploy an auto-encoder architecture as the discriminator. The discriminator loss of BEGAN uses Wasserstein distance to match the distributions of the reconstruction losses of real images with the generated images.

There are also several other objective functions based on Cramer distance [136], Mean/covariance Minimization [137], Maximum mean discrepancy [138], Chi-square [139] have been proposed to improve performance and achieve stable training of GAN.

3.3. Training Tricks:

While research on various GANs architectures and objective functions continue to improve the stability of training, there are several training tricks proposed in the literature intended to achieve excellent training performance. Radford et al. [104] showed using leaky rectified activation functions in both generator and discriminator layers gave higher performance over using other activation functions. Salimans et al. [140] proposed several heuristic approaches which can improve the performance, and training stability of GANs. First, feature matching, changes the objective of the generator to minimize the statistical difference between features of the generated and real images. In this way, the discriminator is trained to learn important features of the real data. Second, minibatch discrimination, where the discriminator process batch of samples, rather than in isolation that helps prevent mode collapse, as the discriminator can identify if the generator continues to generate sample with little variety. Third, historical averaging, that takes the running average of parameters in past and penalizes if there is a large difference between parameters, which can help model to convergence to an equilibrium. Finally, one-sided label smoothing provides smoothed labels to the discriminator instead of 0 or 1, which can smooth the classification boundary of the discriminator.

Sønderby et al. [141] proposed the idea of crippling the discriminator by introducing noise to the samples rather than labels, which prevents the discriminator from overfitting. Heusel et al. [142] used separate learning rate for generator and discriminator, and trained GANs by a Two Time-Scale Update Rule (TTUR) to ensure that model converge to a stationary local Nash equilibrium. To stabilize the training of the discriminator, Miyato et al. [143] proposed normalization technique called spectral normalization.

4. Taxonomy of class imbalance in visual recognition tasks

This section describes different GANs applied to imbalance problem in various visual recognition tasks. We group the imbalance problem in a taxonomy with three main types: 1. Image level imbalances in classification 2. object level imbalances in object detection and 3. pixel level imbalances in segmentation tasks. Understanding this taxonomy of imbalances will provide a valuable framework for further research into synthetic image generation using GAN.

4.1. *Class imbalances in Classification:*

Image classification is the task of classifying an input image according to a set of possible classes. Classification can be broken down into two separate problems: binary classification and multi-class classification. Binary classification involves assigning an input image into one of two classes, whereas in multi-class classification two or several classes are involved. A classic example of a binary image classification problem is the identification of cats or dogs in each input image. Image dataset with high

imbalance, which including inter-class imbalance and intra-classes imbalance, results in poor classification performance.

Inter class Imbalance

Inter-class imbalance refers to a binary image classification problem where a minority class contains smaller number of instances when compared to instances belonging to majority class. Inter class imbalance in a dataset is described in terms of the imbalance ratio. The ratio between the numbers of instances of the majority class and those of the minority class is called the imbalance ratio (IR). For example, binary class imbalance with imbalance ratio of 1:1000 means that for every one-instance in a minority class, there are 1000 instances in the majority class. Datasets with high imbalance ratio is harmful because they bias classifier towards majority class predictions.

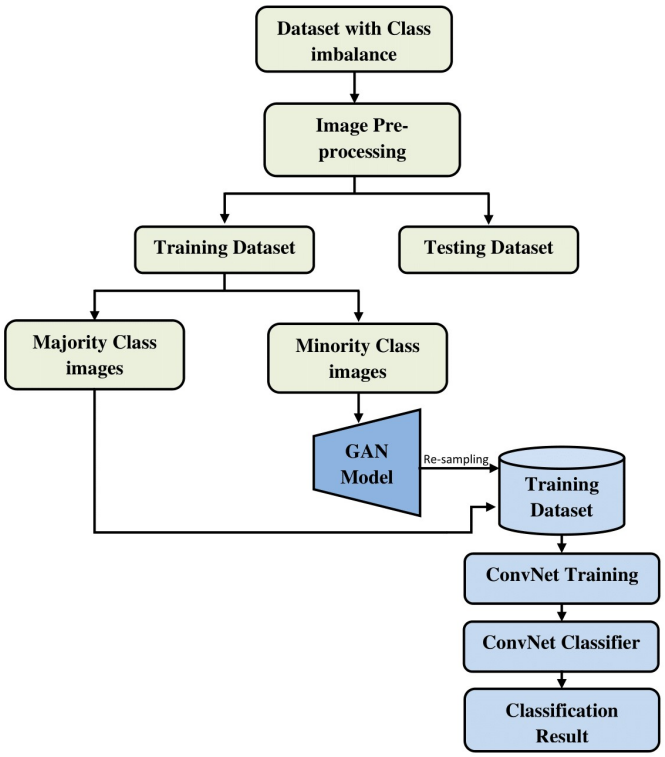

Figure 11: flowchart of GAN-based oversampling technique

Synthetic images generated using GAN can be used as an intelligent oversampling technique to solve class imbalance problems. The general flowchart of GAN-based oversampling technique is depicted in Figure 11. This GAN-based oversampling technique not only increases the representation of the minority class, but it may also help to prevent over fitting.

Shoohi et al. [144] have used DCGAN to restore balance in the distributions of imbalanced malaria dataset. Generated synthetic images from DCGAN are used to achieve 100% balance ratio by oversampling minority class and thus reduce the false positive rate of classification. Their original dataset contains 18,258 cell images, (13,779 parasitized cell, 4,479 uninfected cell). After using imbalanced dataset to achieve 50% accuracy, they observed an increase to 94.5% accuracy once they added the DCGAN-generated samples.

S. Niu et al. [145] introduced surface defect-generation adversarial network (SDGAN), using D2 adversarial loss and cycle consistency loss for industrial defect image generation. SDGAN is trained to

generate defective images from defect-free images. D2 adversarial loss enables the SDGAN to generate defective images of high image quality and diversity, while cycle consistency loss helps to translate defective images from defect-free images. Surface defect classifier trained on the images synthesized by the SDGAN achieved 0.74% error rate and, also proved to be robust to uneven and poor lighting conditions.

Mariani et al. [146] argued that the few examples in minority class may not be sufficient to train GANs, so they introduced a new architecture called Balancing GAN (BAGAN). BAGAN utilizes all available images of minority and majority classes, and then try to achieve class balance by implementing class conditioning in the latent space. Learning useful features from majority classes can help the generative model to generate images for minority classes. An autoencoder is employed to learn an exact class-conditioning in the latent space.

Most of the work done in utilizing GANs based synthetic images for class imbalance and comparing the resulting classification performance have been performed in medical image datasets. In the study of Wu et al. [147], class conditional GAN with mask infilling (ciGAN) is trained to generate examples of mammogram lesions for addressing class imbalance in mammogram classification. Instead of generating malignant images from scratch, ciGAN simulates lesion on non-malignant images. For every non-malignant image, ciGAN generates a malignant lesion onto it using a mask from another malignant lesion. On the DDSM (Digital Database for Screening Mammography) Dataset [148], synthetic images generated using ciGAN improves classification performance by 0.014 AUC over baseline model and 0.009 AUC compared to standard augmentation techniques alone.

The vast majority of studies in bio-medical domain used cycle-GAN [130] to generate synthetic medical images. Muramatsu et al. [149] tested the use of a cycle-GAN to synthesis mammogram lesion images from different organs in mammogram classification. They translated CT images with lung nodules to mammogram lesion images using cycle-GAN and found classification accuracy improved from 65.7% to 67.1% with generated images.

For breast cancer detection, Guan and Loew [150] compared the usefulness of DCGAN-generated mammograms and traditional image augmentation method in a mammogram classification task. On the DDSM Dataset [148],the GAN based oversampling method performed about 3.6% better accuracy than traditional image augmentation techniques.

Most recently, Waheed et al. [151] proposed a variant of ACGAN, called CovidGAN for the generation of synthetic Chest X-Ray (CXR) images to restore balance in the imbalanced dataset. Their dataset contains 721 images of Normal CXR and 403 images of Covid-CXR collected from three publicly accessible databases: 1) COVID-19 Chest X-ray Dataset Initiative [152], 2) IEEE Covid Chest X-ray dataset[153] and 3) COVID-19 Radiography Database [154]. The generator network in the CovidGAN is stacked on top of the discriminator. At the beginning of the training process, the layers of the discriminator are freezed and thus, only generator network gets trained via the discriminator. However, the author offers no explanation for the significance of stacking. They observed improved classification accuracy from 85% to 95% when classifier is trained on combination of original and synthetic images.

The effectiveness of using synthetic images to balance the class distribution is fairly a recent idea that has not been widely tested and understood. At low resolution image datasets, adding synthetic images with original images have shown to improve performance of the classifiers, but at the higher resolution image datasets these synthetic images become obvious to distinguish from the real one. This is due to the fact that the higher resolution images allow for finer textures and details, and hence will need more cautious modifications by GAN so as not to distort the natural patterns occurring in the high-resolution image

dataset. Improving the resolution of GAN samples and testing their effectivenessis an interesting area of future work.

Intra class Imbalance

Another type of imbalance that deteriorates performance of the classification problem is the intra-class imbalances. The techniques used for inter-class imbalance can be extended to intra class imbalance if the datasets have detail labels. However, in real world datasets, data acquisition with detail label is rare because acquiring detailed dataset is costly, and sometimes even not feasible [155]. In many cases, collecting images are tiresome, like 1. capturing images of the same person with glasses and without them, 2. Images of the same person face with varying poses, facial attributes, etc. In some cases, such as the gender swapping, it is not feasible to collect images of the same person as both male and female. Therefore, those techniques for inter-class imbalance are hard to solve intra-class imbalance.

N. Hase et al. [155]presented an interesting idea to combine clustering technique with GANs designed for solving intra class imbalance. The proposed architecture consists of the generator G, the discriminator D, and the pre-trained feature extractor F (Figure 12). The key idea is to generate clusters of images in each class in the feature space, and synthesize images conditioned on class and cluster while estimating the clusters of generated images. The generator G is trained to generate equal number of images for each class and cluster, so that the distribution of both inter and intra class become uniform.

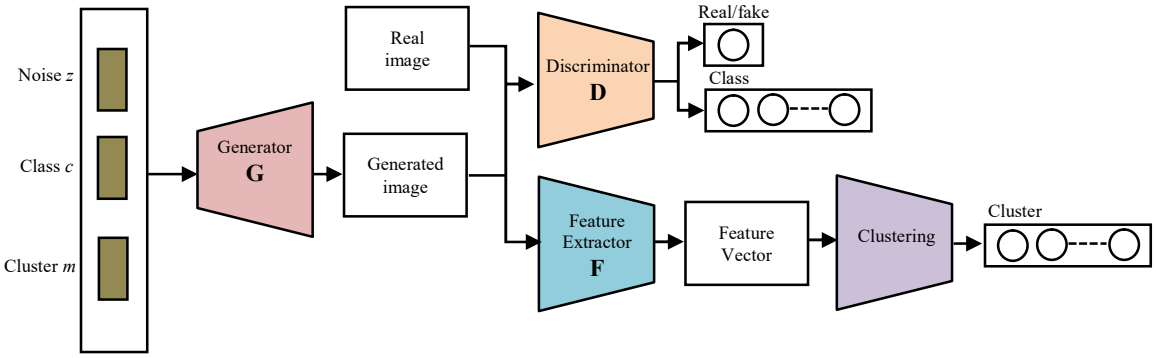

Figure 12: Architecture diagram of clustering based GAN for solving intra-class imbalance presented by N. Hase et al. [155]

Utilizing clustering techniques in the feature space to divide the images into groups for an automatic pattern recognition in the dataset is a promising area for future work. Additionally, it will be interesting to see how the performance of GAN changes with different types of clustering methods such as Hierarchical clustering, Fuzzy clustering, Density-based clustering, etc.

A semantically decomposed GAN (SD-GAN) proposed by C. Donahueet al. [156] adopts Siamese networks that learn to generate images across both inter and intra class variations. Both GANs and Siamese networks have two networks. But unlike GANs, where the two networks compete with each other, the two networks in Siamese networks are similar and working one beside the other. They learn to compare output of the two networks on two different inputs and measure their similarity. For an example, Siamese networks can measure probability that two signatures are made by the same person. A combination of GAN and Siamese networks in SD-GAN can learn to synthesize photorealistic variations (such as, viewpoints, light conditions, scale, backgrounds, and more) of an original input image.

Many studies have reported the problem of an intra-class imbalance owing to age, gender, race and pose attribute variations in a face recognition tasks [157][158][159][160]. Several variants of GAN have been

proposed to address this issue, some focusing on modifying one or more facial attributes, others on generating high quality face images with distinctive pose variations.

Facial Attribute editing:

Human face attributes are highly imbalanced in nature. Attributes can be combined to generate descriptions at multiple levels. For instance, one can describe "white-female" at the category level, or "white-female blond-hair black-eyes wearing necklace" at the attribute level. Attribute level imbalances are inevitable in facial recognition datasets (Figure 13). As an example, Bald persons with a mustache wearing neckties are 14 to 45 times less likely to occur in the CelebA dataset [161].

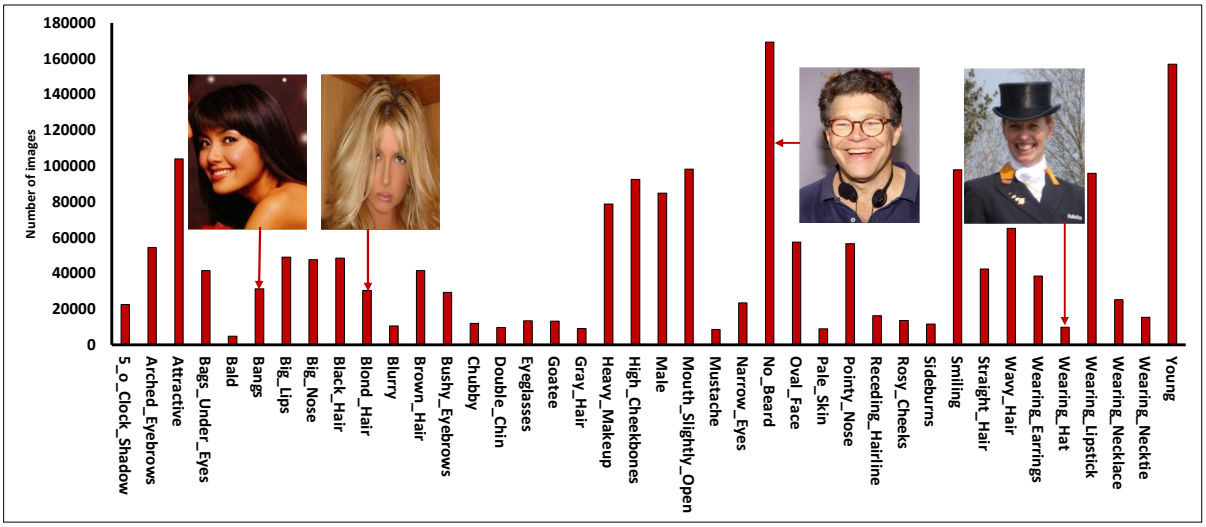

Figure 13: Imbalanced distribution of 40 binary face attributes (positive and negative) on CelebA dataset [161]

Face attribute editing aims to edit the face image by modifying single or multiple attributes while preserving other details. It is challenging because some of the face attributes are locally distributed, such as 'bangs', 'wavy hair', and 'mustache', but some are globally attributed such as 'chubby', 'smiling' and 'attractive'. Several GANs based methods have been proposed to achieve face attribute editing tasks.

Anders et al. [107] proposed a model that combines VAE and GAN together and learn to map the facial images into latent representation. The derived latent representations are then used to find the attribute manipulating direction. For a given facial attribute (e.g., blond hair), the training dataset can be separated to two groups that images with or without blond hair, then the manipulation direction can be computed as the difference between the mean latent representation of two groups. However, such latent representation contains highly correlated attributes, that results in unexpected changes of other attributes, e.g., adding mustache always makes a female become a male as mustache objects are always associated with male in the training set.

Z.He et al. [62] showed how single or multiple facial attributes of a face image can be manipulated by using encoder-decoder architecture. i.e., to generate and modify a face image with the required attributes, while preserving realism of the image. They have introduced encoder-decoder architecture in GAN to handle this task. Encoder in the encoder-decoder architecture maps a facial image onto a latent representation and facial attribute editing is accomplished by decoding the latent representation conditioned on the expected attributes. The authors applied an attribute classification constraint to guarantee that the attributes are correctly edited. Meanwhile, the reconstruction learning is employed to ensure the attribute excluding details are well preserved.

Perarnau et al. [63] proposed an invertible conditional GAN (IcGAN) that is equipped with two encoders to inversely map from input facial images into conditional vector y and latent vector z, which, as a result can be manipulated to generate a new face image with desired attributes. IcGAN is a multi-stage training algorithm that first trains a cGAN[110] to map from conditional vector y and latent vector z to real images, and in a second step learns its inverse mapping from generated images to conditional vector y and latent vector z in a supervised manner (Figure 15). In this way, by changing the conditional vector y, IcGAN allows to control attribute relevant features (e.g. hair color) while latent vector z allows to modify attribute irrelevant features (e.g. pose, background).

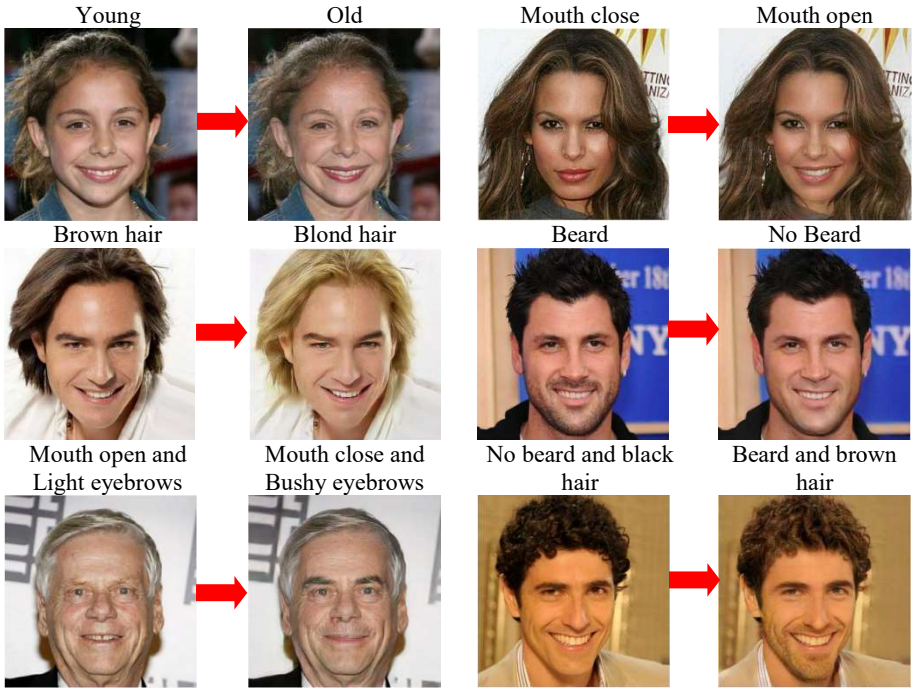

Figure 14: Face attribute editing examples created by AttGAN[162]

Tao et al. [64] argued that the facial attribute editing is an image-to-image translation problem, which aims to transfer facial images from the source domain to the target domain. Their proposed model contains three major parts: an encoder, a decoder, and a residual attributes extractor. The encoder and decoder together constitute generator, whose main aim is to generate a facial image with desired attributes. The encoder maps the facial images into latent representation and the decoder reconstructs (generates) the image from this representation along with attribute vector. The main purpose of residual attributes extractor is to learn the gap between the original input and the desired output in the feature space and back propagate error signal to supervise the generation process.

Zhangi et al.[163] have used the design principle of Adversarially Regularized U-net (ARU-net), instead of conventional encoder and decoder architecture to learn facial attribute editing and generation tasks together during training process. The symmetric skip connection technique is used to pass on the details from encoder to decoder, which preserves the attribute irrelevant features. In this architecture, the ARU-net is integrated with GANs that results in ARU-GAN to perform facial attribute editing. The ARU-GAN consists of four major components: the ARU-net for preserving attribute irrelevant features, the adversarial network to constrain the latent representation, the discriminator to distinguish between real and fake image, and the attribute classifier to ensure the desired attributes are edited.

Zhang et al. [164] introduced a spatial attention mechanism into GANs for only modifying attribute relevant part and keep attribute irrelevant part unchanged. SaGAN [133] is used to locate and manipulate

attribute-relevant part more precisely. The generator of the proposed architecture consists of an attribute manipulation network (AMN) and a spatial attention network (SAN). Given a face image, SAN learns to localize the attribute-specific region and then AMN edit the face image with the desired attributes in the specific region located by SAN.

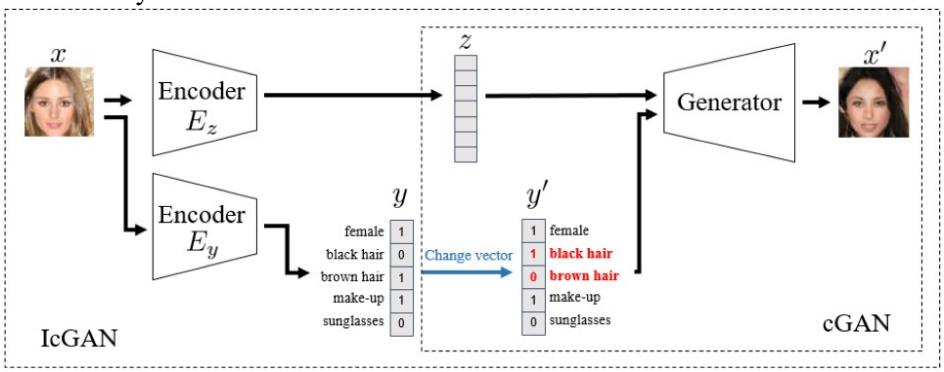

Figure 15:illustration of invertible conditional GAN presented by Perarnau et al. [165]

The major downside with the current approaches is that the input to GAN should be frontal face images. It will be interesting to explore a new architecture that can be trained to modify the attributes of side-view or any arbitrary views.

Person re-identification:

Person re-identification [166] is another challenging task worth mentioning, which are adversely affected due to significant intra class imbalance. Intra class variations caused by rotation (varying poses) are often larger than the inter-person dissimilarities used to differentiate the face images [167]. Recent face-recognition surveys [168][169] identified pose variation as one of the prominent unresolved issues in face-recognition task. For instance, in order to maintain the highest standard of security, smart video system needs to be able to detect person invariant to pose (Figure 16).

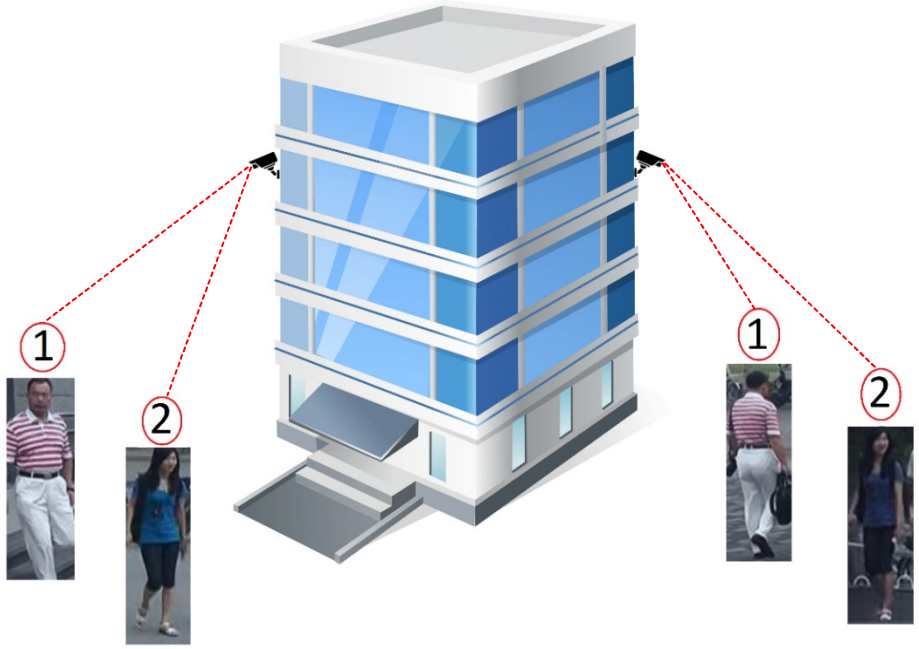

Figure 16: Example of Person reidentification task. Person reidentification is a key element in video surveillance that deals with matching images of same person over many non-overlapping camera views.

Qian et al. [170] introduced a pose-normalized GAN model (PN-GAN) for alleviating the effects of pose variation. Given any pedestrian image and a desirable pose as input, the model utilized a desirable pose to produce a synthetic image of the same identity with the original pose replaced with the desirable pose (Figure 17). After this, the authors trained the re-identification model with the original images and generated pose-normalized images to extract two set of features. Finally, they fused the two types of features as the final feature. As a result, the features extracted from the synthesized images improved generalization ability of re-identification model.

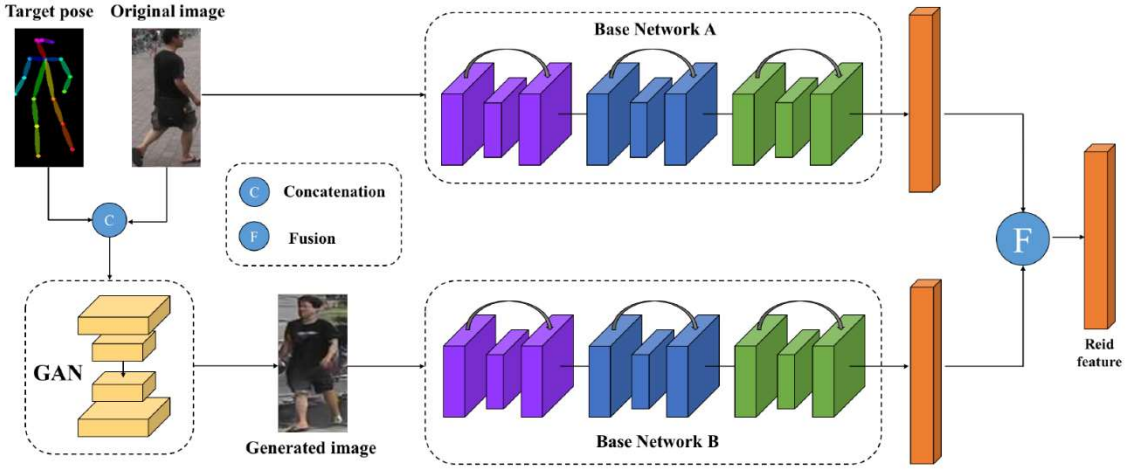

Figure 17: Architecture diagram of pose-normalized GAN presented by Qian et al. [170]

To address person re-identification challenges in complex scenarios, Wei et al. [171] proposed a model called Person Transfer Generative Adversarial Network (PTGAN) for implausible person image style transfer from source domain to target domain, across datasets with different styles, such as backgrounds, poses, seasons, lightings, etc. The domain transfer procedure in PTGAN is inspired by CycleGAN [130]. Different from Cycle-GAN [130], PTGAN incorporates additional constraints on the person foregrounds to make sure the stability of their identities during transfer. Compared with Cycle-GAN, PTGAN generates high resolution person images, where person identities are unchanged, and the styles are transformed.

Being a cross-camera tracking and human retrieval task, person re-identification often suffers from image style variations resulting from different cameras. Therefore, Zhong et al. [172] designed a camera style adaption model for adjusting ConvNet training. They have used CycleGAN [130] for transferring images from one camera to the style of another camera. Given that both original and style transferred images, identification discriminative embedding (IDE) is used to train the ConvNet model. Particularly, authors have used ResNet-50 pre-trained on ImageNet dataset as backbone and follow the fine-tuning strategy.

Pedestrian images suffer from information loss when transferring from one camera to the style of another camera. Deng et al. [173] presented a model, named similarity preserving cycle consistent generative adversarial network (SPGAN), which is composed of a CycleGAN and a Siamese network (SiaNet). CycleGAN learns to translate pedestrian images from one domain to other domain, and the contrastive loss induced by the SiaNet pulls close a translated image and its counterpart in the source domain, and move away the translated image and any image in the target domain.

Ge et al. [174] presented a Feature Distilling Generative Adversarial Network (FD-GAN) that aims at learning identity related and pose-unrelated person representations. The proposed model adopts a Siamese structure with multiple novel discriminators on human poses (pose discriminator) and identities (identity

discriminator). The idea behind FD-GAN is to learn pose-unrelated and identity-related features of pedestrian image, then it can be used to generate the same pedestrian image but with different target poses.

Although the existing GAN-based methods have achieved excellent performance in image-based person re-identification, it still needs considerable effort to tackle the video-based identification datasets. Future work seeks to expand to use GAN for generating a sequence of images for the video-based identification datasets.

Vehicle Re-identification:

Vehicle Re-identification task is even more challenging as it suffers from large intra-class differences caused by viewpoint and illuminations variations, and inter-class similarity primarily for different identities with the similar look (Figure 18).

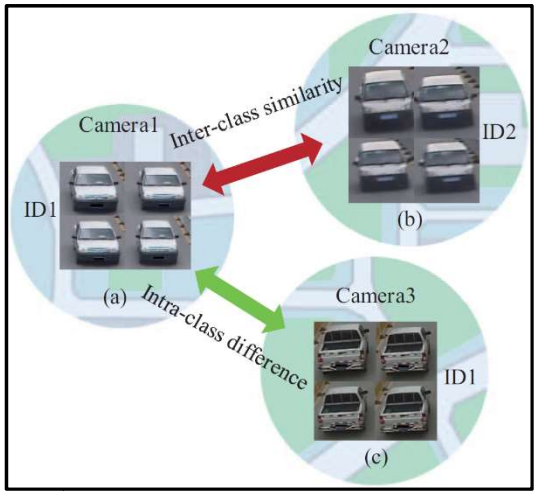

Figure 18:illustration of challenges in vehicle Re-identification provided by Zheng et al [175].

Y. Zhou et al. [176] proposed a model called Cross view GAN to generate images in different viewpoints of the same vehicle. Cross view GAN composed of classification, generator, and discriminator network. First, classification network is trained to learn vehicle intrinsic features such as model, color, and type information. In addition to intrinsic features, it also learns viewpoint features. Then the generative network is conditioned on the average feature of the expected viewpoint and vehicle's intrinsic features to infer images of the same vehicle in other viewpoints. The discriminator network learns to distinguish real images from the generated images, while ensuring images are generated with correct attributes.

F. Wu et al. [177] improve the discriminative power of ResNet-50 model for the Vehicle re-ID task by simultaneously training with initial labeled images and DCGAN generated unlabeled images. They further explore the effectiveness of using DCGAN generated images on a wide range of vehicle re-ID datasets and show improved performance of vehicle re-identification.

Fine-grained image classification:

The fine-grained image classification is also attributed to major variations in the intra-class and minor inter class variations [178]. It is a difficult task for two reasons. First, the training samples of each class are inadequate. Second, the differences between different classes of images are quite small [179]. As an example, it is very difficult to identify the images of Shetland Sheepdog from that of Collie dog. Similarly, the images of Sayornis and Gray Kingbird are quite difficult to distinguish (Figure 19).

Figure 19: Sample images from the Stanford Dogs dataset [180] *and the Caltech-UCSD Birds dataset* [181]*, which exhibits minor inter-class variations and major intra-class variations.*

Y. Fu et al. [178] developed a model called Fine grained conditional GAN (F-CGAN) to solve fine grained class dependent image synthesis problem. F-CGAN consists of three main components: 1. a 2-stage GAN, 2. a fine-grained feature preserver and 3. a multi-task classification model. The 2-stage GAN generates high resolution images, the fine-grained feature preserver targets to capture fine grained details and the multi-task classification model utilizes generated image data to improve fine grained classification accuracy.

C. Wang et al. [182] find that the discriminator in GANs learns a hierarchical identification features of the fine-grained classes and discriminate pattern of the fine-grained training samples. They use the architecture pictured below to implement the fine-grained Plankton classification task (Figure 20). The main idea is to train a fine-grained classifier that share weights with discriminator of the DCGAN, which force discriminator to concentrate on features of small classes. On WHOI-Plankton dataset [183], F1 score of the classifier improved by over 7%.

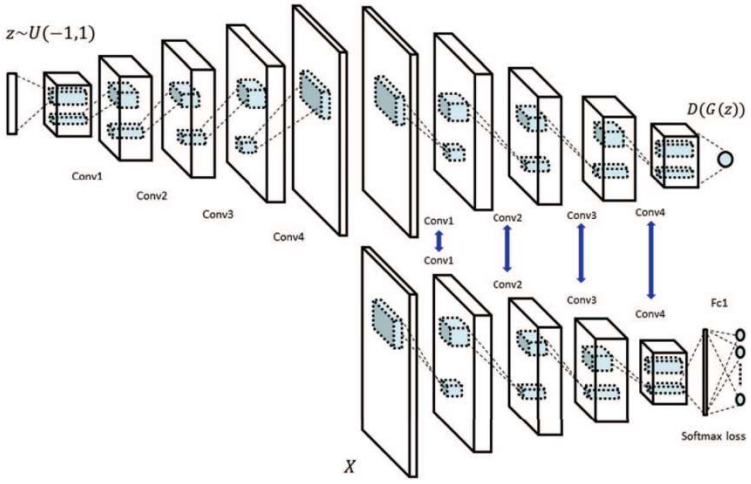

Figure 20:Complete fine-grained Plankton classifier architecture used by C. Wang et al. [182]

Typically, medical image datasets contain both general labels, e.g., “male”, “female” and disease specific detailed labels. It is mentioned that the complexity and nature of data is hard to learn by using single GAN. Hence, T. Koga et al.[184] connected two GANs in series, one for learning general features and other for detailed features. The first GAN generates diverse images, which takes a noise vector and general labels as inputs. The second GAN receives synthetic image generated by the first GAN, and disease specific detailed labels as inputs, and generates the final fine-grained medical images.

Multiclass Imbalance

In many real world problem such as emotion classification [185], plant disease classification [186], medical image classification [187], industrial defect classification [188] etc., it is more likely that more than one class exists and needs to be recognized. Multiclass classification have been shown to suffer learning difficulties than binary class classification, because multiclass classification increases the data complexity and intensify the imbalanced distribution [189]. Three types of imbalance could occur to the multiclass datasets: few minority-many majority classes, many minority-few majority classes, and many minority-many majority classes. Shuo Wang et.al [190] studied impact of all different types of multiclass imbalances and showed that they negatively affect minority class and overall performance.

An example of few minority-many majority class imbalance is an emotion classification, as some classes of emotions like disgusted are relatively uncommon compared to common emotions like happy or sad. Zhu et al. [191] employed cycle-GAN which can synthesis uncommon emotion class like disgusted from the frequent classes (Figure 21). In addition to adversarial and cycle consistency loss, they use least square loss from LSGAN to avoid vanishing gradient problem. Employing cycle-GAN based data minority class data augmentation achieved 5–10% increase in the overall accuracy. They also found that enlarging minority class also increases accuracy of other majority classes.

Figure 21: On emotion classification task [191]*, the images on the left are original data and the rest are images generated by cycle-GAN.*

Weather Image classification is another example of few minority-many majority class imbalance, because some type of weather, like snow, is relatively rare compared to sunny, haze and rainy days. Li et al. [192] used DCGAN to generate images of minority classes in training. They found that the GAN-based data augmentation technique led to margin clarity between classes and hence improvement in classification performance.

Y. Huang et al. [193] presented an interesting idea to combine ensemble learning with GANs designed to address the class imbalance problem in weather classification. The proposed method comprised of three ingredients as depicted in (Figure 22): 1. DCGAN to generate synthetic image and balance the training dataset 2. Nearest neighbor method to remove any possible outlier images generated by DCGAN 3. an ensemble learning method to combine the classification results of the multiple classifiers so as to achieve better results.

The use of DCGAN was tested by Salehinejad et al. [187] in the task of chest pathology classification. Using chest X-ray images, they build a deep ConvNet classifier to classify 5 different anemic classes. Their dataset is highly imbalanced, contains three majority and two minority classes (Figure 24a). The synthetic images generated using DCGAN were used to balance and augment the original imbalanced dataset. They demonstrated that a combination of the original imbalanced dataset and generated images improves the accuracy of deep ConvNet classifier in comparison to the same classifier trained with original imbalanced dataset alone. On chest X-ray dataset [187], a mean classification accuracy improved from 70.87% to 92.10%.

Frid-Adar et al. [194] also showed that generating synthetic liver lesion images using DCGAN can improve classification results. They combined standard augmentation technique and DCGAN generated synthetic images to train a classifier. Their liver lesion dataset contains 182 computed tomography images (65 hemangiomas, 64 metastases and 53 cysts). By adding the synthetic images to standard data augmentation, their classification performance increased from 78.6% sensitivity and 88.4% specificity using standard augmentations to 85.7% sensitivity and 92.4% specificity using DCGAN-based synthetic images.

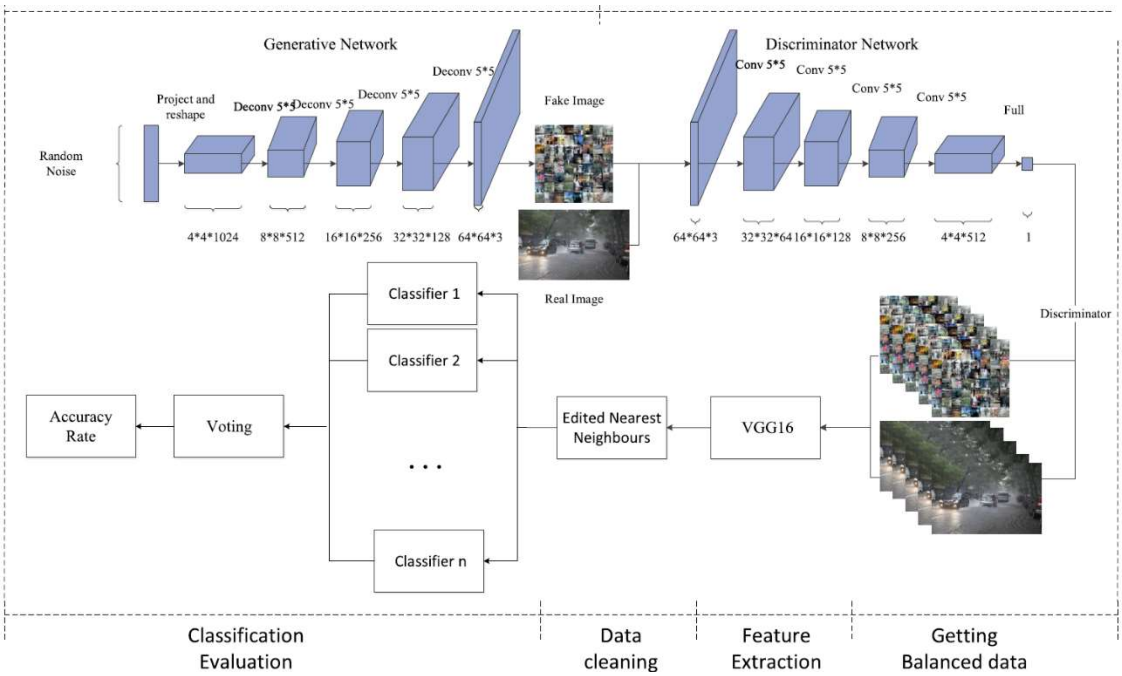

Figure 22: Illustration from Y. Huang et al. [193] *showing how the Ensemble learning is integrated with GAN Framework.*

H. Rashid et al. [195] tested effective of using GANs to generate skin lesion images. Using ISIC 2018 dataset [196], Dermatoscopic image database, they build a CNN classifier to classify 7 different skin lesion as depicted in Figure 23. These classes are highly imbalanced, and the GAN is used as a method of intelligent oversampling.

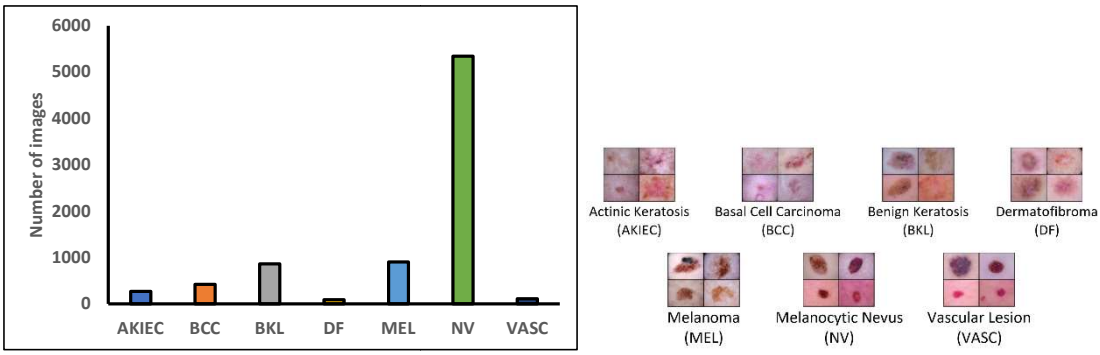

Figure 23: (a) Distribution of the seven skin lesion class labels of the ISIC 2018 dataset [196]. *(b) Sample images from each class.*

Nazki et al. [186] used Cycle-GAN to alleviate multiclass imbalance problem in tomato plant disease classification. Their tomato plant disease dataset contains 2789 images, highly suffered from class imbalance in 9 disease categories (Figure 24b). Using Cycle-GAN, they translated images from the healthy tomato leaves to underrepresented diseased tomato leaves. This study demonstrated that the synthetic image generated by Cycle-GAN can be used as an augmented training set to improve the performance of classifier.

Bhatia et al. [197] sought out to compare synthetic image generated using WGAN-GP against the standard data augmentation in the context of multiclass image classification. They artificially introduced class imbalance in two balanced datasets of CIFAR-10 [198] and FMNIST [199], and studied the effects of multiclass imbalance on classification performance. On CIFAR-10 [198] dataset, classification performance improved from 80.84% accuracy and 0.806 F1-score using standard data augmentation to 81.89% accuracy and 0.812 F1-score using WGAN-GP. On FMNIST [199] dataset, performance improved from 91.9% accuracy and 0.921 F1-score using augmentation to 92.8% accuracy and 0.923 F1-score using WGAN-GP.

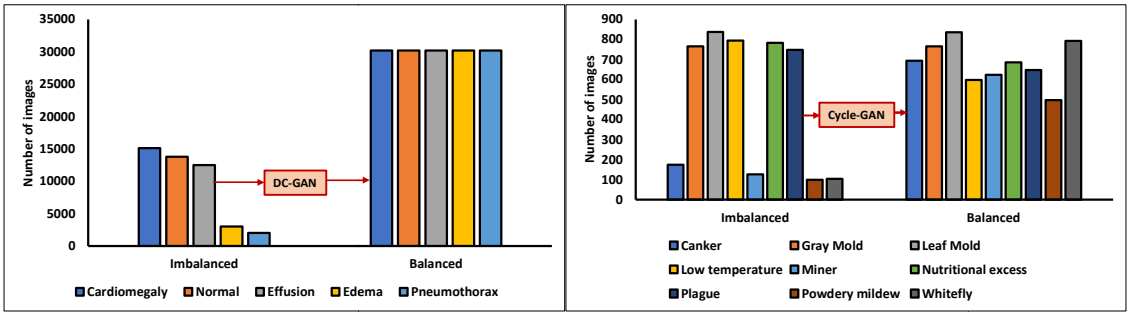

Figure 24: The distributions of (a) Chest X-ray image dataset and (b) tomato plant disease dataset, before (left) and after class balancing using GANs (right).

An idea of GANs based transfer learning technique for multiclass imbalance problem is proposed by Fanny et al. [200]. Their architecture named class expert generative adversarial network (CE-GAN) makes use of multiple GANs models, a separate GANs for each class. Feature maps in the main classifier are arranged in parallel, with each feature maps pretrained to identify the characteristics of a single class in the training data (Figure 25). The weights of the pretrained feature maps are transferred from discriminators of the GANs to main classifier model for further training in a supervised mode.

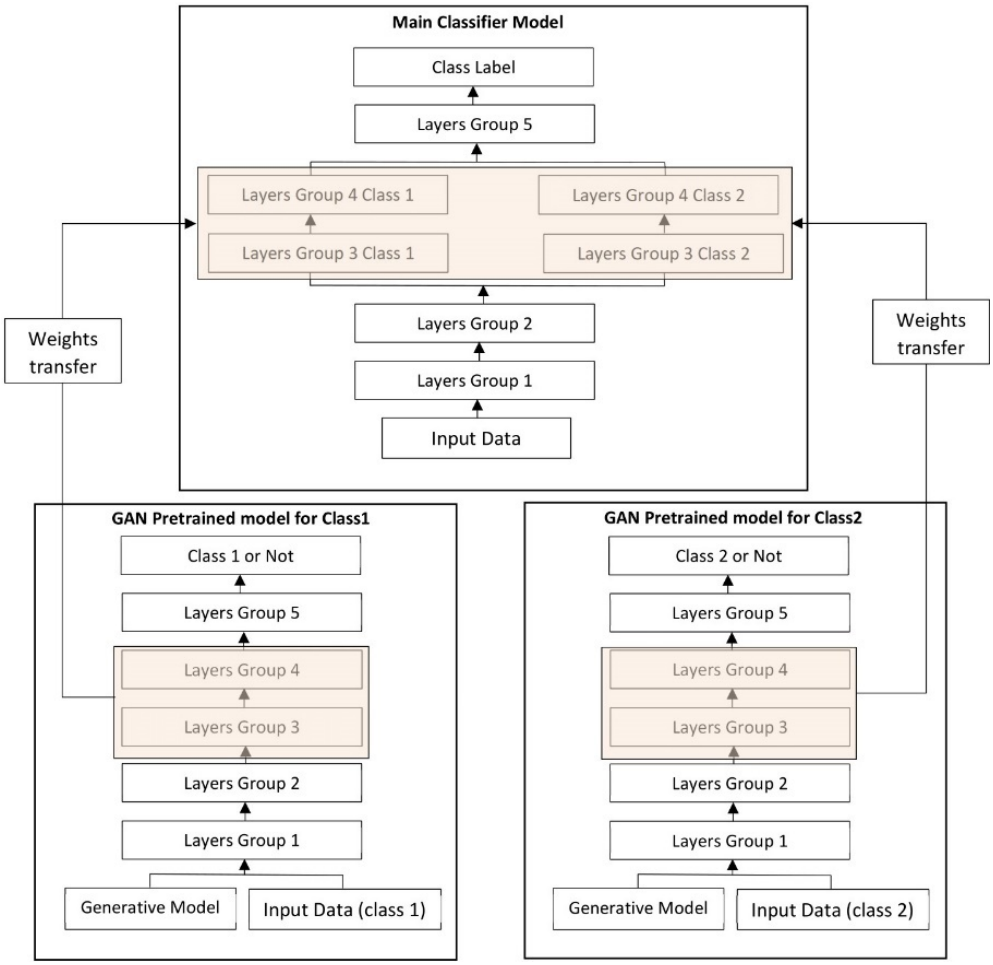

Figure 25: Illustration of the class expert generative adversarial network architecture [200]

The GAN-based synthetic images served as an intelligent oversampling technique and can address the problem of multi-class imbalance to a greater extent. However, synthetic images must be used with caution because if the quality of the synthesized images is not high, this would lead to additional noise to the original datasets.

4.2. Object level imbalances in Object Detection:

Object-Scale Imbalance

One pervasive challenge in the scale invariant object detection is large scale variance across object instances, and particularly, detecting small objects are more challenging than medium and large-scale objects. As per MS COCO definition [201], Objects with size less than 32×32 pixels are small, size between 32×32 to 96×96 pixels are considered as medium and objects with size greater than 96×96 pixels are large objects (Table 2). On the one hand, small objects in MS COCO dataset accounts for only 1.23% of total object area, on the other hand, medium and large-scale objects are over 98% of object area. Object detection algorithm should be able to detect both small objects as well as medium and large objects. Detecting small objects are essential in many real-world applications. For instance, detecting distant or small objects in the high-resolution driving scene images captured from car is essential for achieving autonomous driving. Many distant objects, such as traffic lights or cars, are imperceptible as shown in Figure 26. Haoyue et al. [202] measure the extent of scale variation using the coefficient of variation

(CV), determined as the ratio of the standard deviation to the mean of the object scale. The bigger the CV, the more complicated the problem of scale variation.

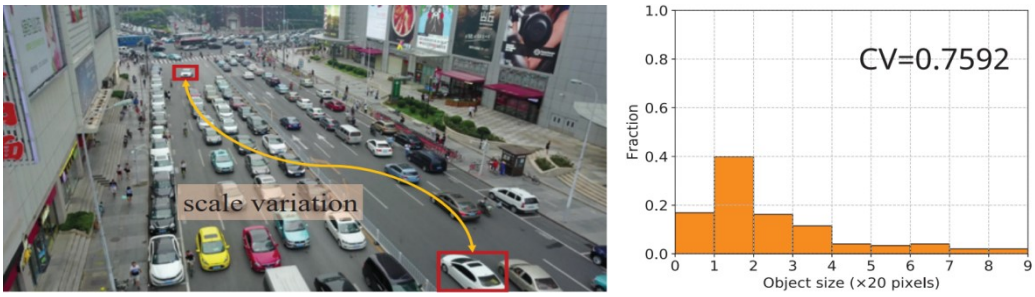

Figure 26: Example of scale variation and the scale (object size) distribution of the VisDrone2019 dataset objects in pixels [202].

There can be three reasons why detecting small objects are more complicated than larger one: 1. Small objects occupy a much smaller area, and consequently there exists lack of diversity where small objects are located in the image, 2. There are comparatively less images in the dataset containing small objects which may bias any object detection algorithm to concentrate more on medium and large-scale objects, and 3. The activations of small objects become smaller and smaller with each pooling layer in a standard convNet architecture as it progressively reduce the spatial size of an image.

Object Category	**Spatial dimension**		**Object Count**	**Total Object Area**
	Minimum	**Maximum**		
Small	0×0	32×32	41.43%	1.23%
Medium	32×32	96×96	34.32%	10.18%
Large	96×96	∞×∞	24.24%	88.59%

Table 2: The definitions and statistics of the small, medium, and large objects as MS COCO [201]

To overcome the problem of scale imbalance, two different strategies based on GAN have been proposed in the literature. Commonly adopted strategy is to convert low resolution small object features into high resolution features [203] using GAN. Diversity of the small object locations in the images are enhanced by copy-pasting small object instances several times in each image through adversarial process [204].

Li et al. [203] utilized a GAN framework that transforms poor representation of small-scale objects to super-resolved large object. The generator attempts to generate super resolution features for the small objects. The discriminator in this framework is decomposed into two branches, namely, a perceptual branch and an adversarial branch. An adversarial branch is trained to discriminate between real large-scale object and generated super resolution object while a perceptual branch helps to make sure that the generated super-resolved object is useful for the detection. They tested the effectiveness of this framework on Tsinghua-Tencent 100k dataset [205], PASCALVOC dataset [206] and Caltech pedestrian benchmark [207].On the PASCAL VOC 2007 dataset [206], The Average precision (AP) of small objects such as plant, chair, bottle and boat increased by 10%, 15.1%, 21.9% and 10% respectively, compared with Faster-RCNN.

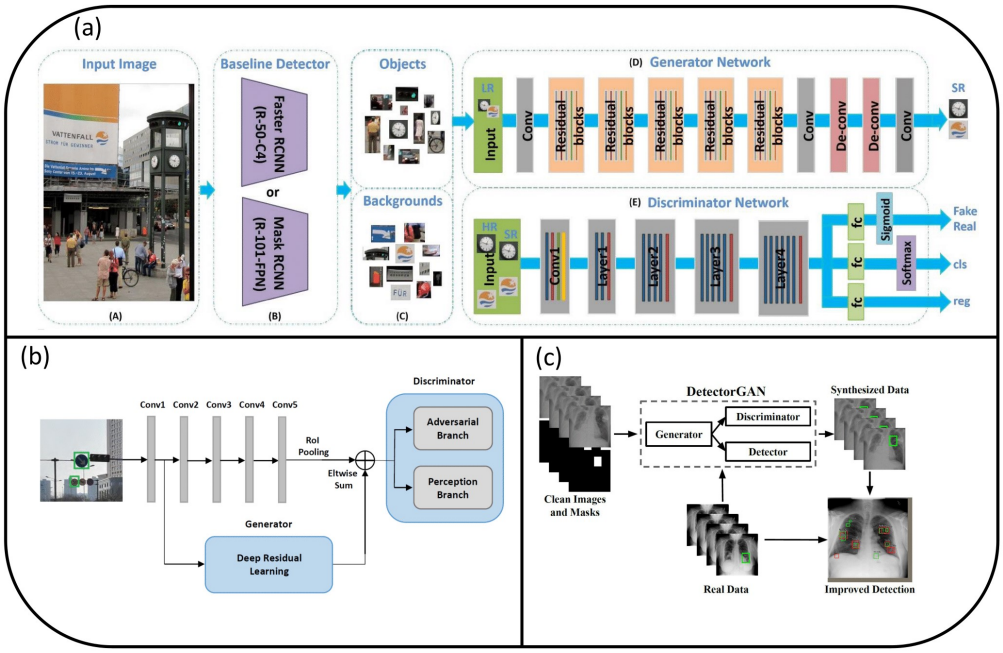

Figure 27: Architecture diagram of (a) SOD-MTGAN [208] *(b) Perceptual GAN* [203] *and (c) Detector GAN* [204].

Bai et al. [208] Used baseline detectors such as Faster RCNN [36], Mask RCNN [209] to crop an input image into smaller regions (generate ROIs) and then use generator network to reconstruct up-scaled version (super resolved) of cropped regions, while the discriminator perform multiple tasks that discriminates the real from the high resolution generated images, perform classification and regress the bounding box co-ordinates (object location) simultaneously.

Lanlan Liu et al. [204] proposed a Detector GAN that combine and optimize both GANs and object detector together. The generator is trained with both adversarial and training loss, which generates multiple small objects in an image that are hard to detect by the detector and hence enhance the robustness of the detector.

Imbalance due to occlusions and deformations

Like the object scale imbalance, occluded and deformed objects in the images follow a skewed distribution. For instance, occlusion from other cars due to urban traffic or parking lot is more common than from an air conditioner as shown in Figure 29. The performance object detection is often suffered from imbalance due to occluded and deformed objects. Zhu et al. [210] define occlusion ratio to measure the degree of occlusion, determined as the fraction of pixels being occluded. As per VisDrone-DET2018 dataset [210], objects with occlusion ratio greater than 50% are heavy occlusion, ratio between 1% to 50% are considered as partial occlusion and objects with 0% occlusion ratio are categorized as no occlusion. The bar chart below (Figure 28) depicts the imbalanced distributions of occluded, partially occluded and heavily occluded objects in VisDrone-DET2018 dataset [210].

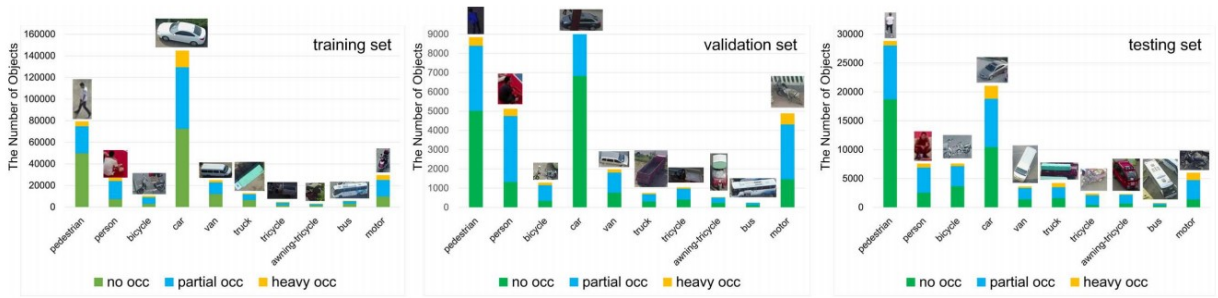

Figure 28:Imbalanced distribution of occluded, partially occluded and heavily occluded objects in VisDrone-DET2018 dataset [210]

One way to build the robust object detector invariance to occlusion and deformation is to generate realistic images of these rare occurrences using GANs, and then train object detector with the generated images. Adversarial object detection could be another interesting way to generate all possible occlusions or deformations on the feature maps that make recognition hard. The object detector is simultaneously trained to overcome the difficulties imposed by the adversarial task.

Figure 29: illustration of real world occlusions and deformations provided by Wang et al. [211]

Wang et al. [211] utilized the adversarial spatial dropout to simulate all kind of rare deformations and occlusions on the feature maps that are hard for the object detector to detect. Unlike traditional methods [47] that add occlusions on foreground objects in pixel space, they focused on feature space. Their architecture (Figure 30) comprised of two networks: Adversarial Spatial Dropout Network (ASDN) and Adversarial Spatial Transformer Network (ASTN) to create occlusion and deformation respectively. On VOC2007 and VOC2012 datasets, this architecture achieved increase in mean Average Precision (mAP) of 2.3% and 2.6% respectively compared to the Fast-RCNN [36].

Inspired by this architecture, Chen et.al [212] proposed Adversarial Occlusion Aware Face Detection (AOFD) to overcome the problem of limited occluded face image in training dataset. As opposed to cropping or erasing, Dwibedi et.al [213] utilized GAN to insert new objects on the images by cut and paste. This method can be extended by inserting occluded and deformed objects on the training images.

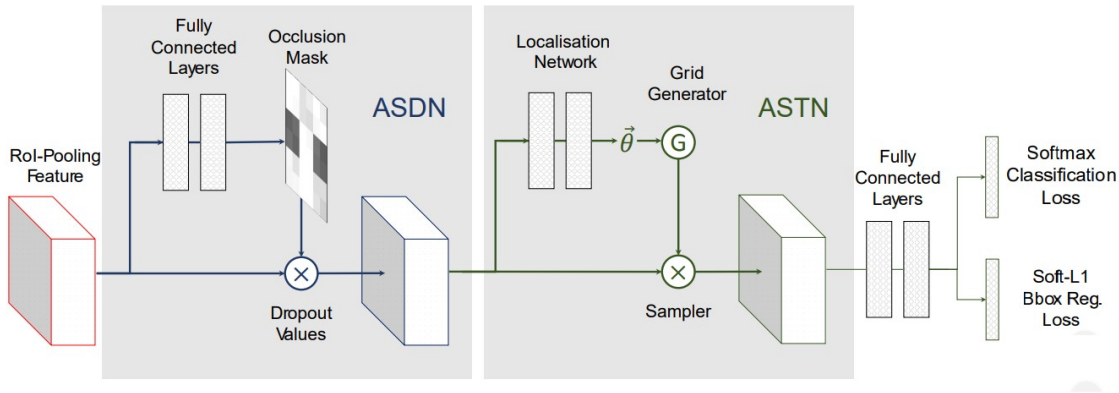

Figure 30: Architecture diagram to generate hard examples for training [211]

Taking full advantage of GANs and combining them into different ConvNet architectures is a recent trend in object detection. This kind of architectures are often called as a three-player GAN. In an attempt to improve performance of detection and classification, three-player GAN only generates hard-to-classify samples. Particularly, the use of faster R-CNN with GANs has improved the state of-the-art benchmarks. Testing the performance of different combinations in comparisons to current state of-the-art models is an interesting area for future work.

Foreground-background object class imbalance

Both single stage and two stage object detection algorithms evaluate multiple regions in an image during the training stage. But only a few regions contain foreground (positive), the rest are background (negative). Many of the background examples are easy to classify and offer an uninformative training signal. Just a few background examples provide rich information for training. The imbalance between foreground (objects) and easily classified background overwhelms cross entropy loss and gradients from converging. Some form of hard sampling is a commonly used method by the object detection algorithms to account for this imbalance. The most straightforward and simple hard sampling method is uniform random sampling that randomly selects a subset of negative and positive examples (uniformly distributed) for evaluation. Hard negative mining is another hard sampling method that selects hard samples as negative examples instead of random selection to improve the detection performance.

Unlike hard sampling methods, GAN address the problem of foreground background imbalance by directly injecting hard positive and negative synthetic examples into the training dataset. Task aware data synthesis proposed by Tripathi et.al [214] uses GAN based approach to generate hard positive examples that improve the detectors classification accuracy. Their architecture utilizes three competing networks (Figure 31): a synthesizer (S), a discriminator (D) and the target network (T).Given a background image and a hard-positive foreground mask, synthesizer aims to optimally paste foreground mask onto the background image to produce a realistic image that can fool both the target and discriminator networks. The discriminator network provides necessary feedback to the synthesizer which ensures the realism of generated composite image. The target network is a pre-trained object detector such as SSD and faster R-CNN. On the VOC person detection dataset, this architecture achieved a performance improvement of up to 2.7%.

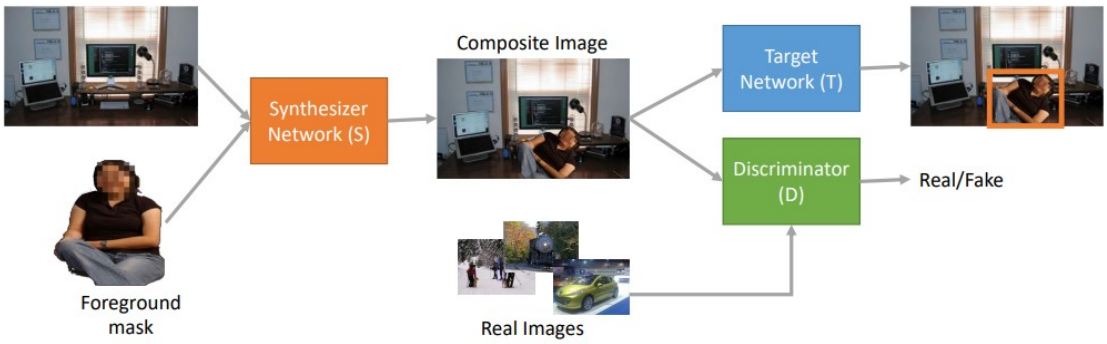

Figure 31:Pipeline of task aware image synthesis used by Tripathi et.al [214]

H. Wang et al.[215] presented an interesting idea of object detection via progressive and selective instance-switching (PSIS). Given a pair of training images, PSIS synthesize a new pair of images by swapping objects of same class between an original pair of images by also considering scale and shape information of the objects. Generating more training images by swapping objects of low-performing classes improves overall detection accuracy.

Gene-GAN [216] proposed by S. Zhou et al. employ an encoder and a decoder architecture to replace an object in an image with different object from a second image. Given an image, Encoder decomposes it into the background and object feature vectors, while decoder reconstructs a new image by transplanting an encoded object to it.

4.3. *Pixel level imbalances in Segmentation*

Pixel-wise class imbalance

GANs are being employed to solve pixel level class imbalance problem in segmentation tasks that have a negative influence on segmentation accuracy. The use of image to image translation GANs for a pixel-level augmentation on segmentation tasks was tested by S. Liu et al. [217] . Particularly, they used Pix2pix HD GAN [135] to translate semantic label maps to realistic images. Semantic object labels from the original dataset such as street, car, pedestrian etc. are recombined to synthesis new label maps which can balance the semantic label distribution. Then the new balanced label maps are translated to realistic images by Pix2pix HD GAN. To further understand the effectiveness of this method, a study was conducted by balancing one to many label classes on original label maps. On the Cityscapes dataset [56] this resulted in an improved mean accuracy of a specific class up to 5.5% and the average overall segmentation accuracy up to 2%.

Shadow detection is a segmentation problem in which there are substantially lesser shadow pixels than non-shadow pixels in training images. V. Nguyen et al. [218] presented Sensitivity conditional GAN (ScGAN), an extension of cGAN[110], tailored to tackle the challenging problem of pixel-level imbalance. To balance shadow and non-shadow pixel imbalance during training process, Sensitivity parameter W is introduced in ScGAN that controls how much to penalize the false positive prediction. Notably, Sensitivity parameter W is made tunable by allowing it to interact with generator in addition to loss function (Figure 32). ScGAN achieved up to 17% error reduction on UCF [219] and SBU [220] dataset with respect to the previous state-of-the-art model.

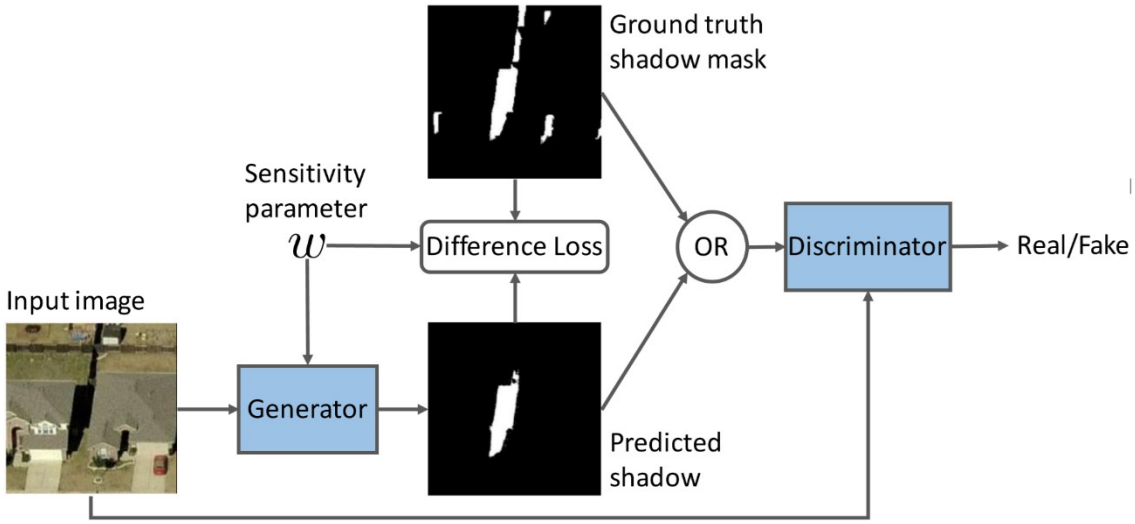

Figure 32:Illustration of Sensitivity conditional GAN[218]

Voxel GAN architecture proposed by M. Rezaei et al. [221] is a 3D GAN model to address the pixel level imbalance problem in the brain tumor segmentation task as the majority of the pixels belongs to the healthy region and only few pixels belongs to tumor region. Voxel GAN is made of 3D segmentor network to learn generating segmentation labels from 3D MRIs, and a discriminative network to differentiate generated segmentation labels from real labels. The segmentor and discriminator are trained by mix of adversarial loss with weighted $\ell 1$ loss and weighted categorical cross-entropy loss to reduce the negative impact of pixel imbalance.

Similar to this work, M. Rezaei [222] used similar loss function by mixing adversarial loss and weighted categorical accuracy loss to handle imbalanced training dataset of whole heart segmentation task. Balancing through ensemble learning by combining two discriminator to improve their generalization ability of the GAN was tested by M. Rezaei et al. [223] in medical image semantic segmentation task. One discriminator classifies whether generated segmentation label is real or fake. Another discriminator is trained to predict false positives and false negatives. Final segmentation mask is generated through adding the false negatives and removing the false positives predicted by this discriminator.

Imbalance due to occlusions in segmentation

GANs are also very efficient in segmentation of natural settings with severe occlusion and large-scale changes [224]. Sa et al. [225] describe that occlusion is a key challenge in segmenting dense scenes. Objects in dense scenes often occlude each other, which lead to severe information loss. In many cases, segmentation algorithms cannot infer appearance of the objects beyond their visible parts, which may prevent it from making accurate decisions if a person purposely covers the face. GANs offer a new way to generate the invisible parts of objects, i.e., learns to complete the appearance of occluded objects.

SeGAN [226], developed by Ehsani et al., is an interesting framework to segment the invisible part of the object and then generate the appearance by painting the invisible parts. The proposed framework uses a segmentor, a generator, and a discriminator to combine segmentation and generation tasks (Figure 33). The segmentor takes an image and segmentation mask of the visible region of an object as an input, and then predicts an intermediate mask of the entire occluded object. The generator and discriminator are trained to generate an object image in which the invisible regions of the object are reconstructed.

Dong et al.[227] proposed a two stage model, named Occlusion-Aware GAN (OA-GAN), to remove arbitrary facial occlusions, e.g., faces with mask, microphone, cigarette, etc. OA-GAN is equipped with

two GANs: The first GAN G_1 is designed to disentangle the occlusion, and the second GAN G_2 is trained to generate the occlusion free images given the generated occlusions.

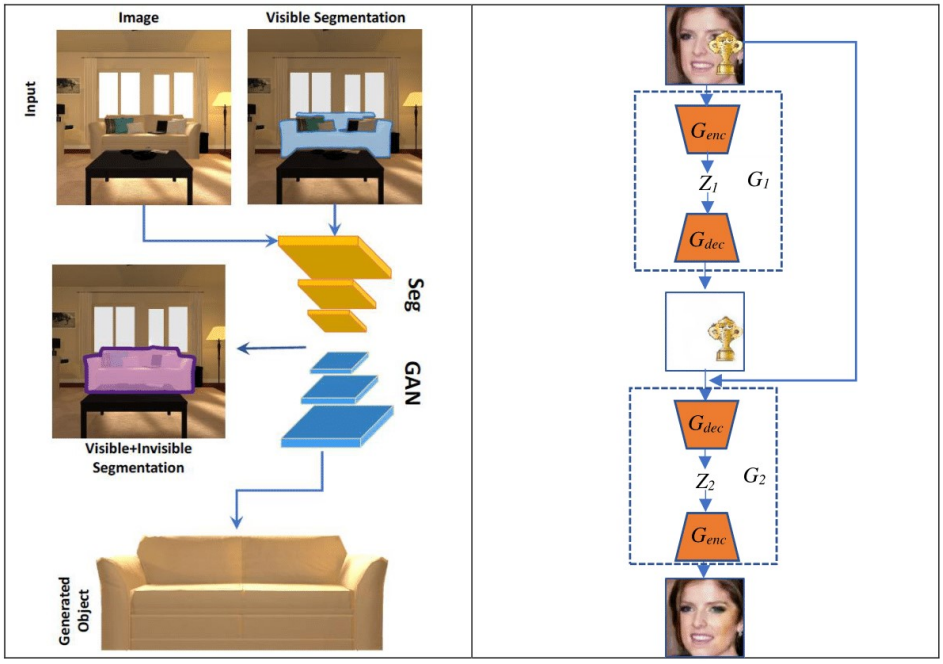

Figure 33:Illustration of SeGAN[226]*(left) and Occlusion-Aware GAN* [227]*(right)*

5. Discussion

To provide a detailed overview and better comparison of various studies for imbalances in computer vision, the surveyed works have been summarized in Table 3. GANs based methods that address the imbalance problem in classification tasks aim to increase the classification accuracy for the minority classes. Many of these methods use image-to-image translation to generate minority class images from one of the majority classes, while others generate minority class images from the random noise vector. GANs based intelligent oversampling method outperforms both traditional sampling and data augmentation methods in classifying imbalanced image data. However, it is not clear that how much synthetic images must be blended with original images to achieve the maximum performance of the classifiers. Additionally, synthetic images would lead to additional noise to the original training dataset if the quality of the synthesized images is poor. Therefore, most of the surveyed methods in GANs based intelligent oversampling methods focused mainly on balancing distribution as well as improving quality of the generated images.

Image-to-image translation methods used for inter-class imbalance problem cannot be extended to solve intra class imbalance as it is difficult to acquire image datasets with detail labels. The interesting way to solve this problem is to employ clustering techniques in the feature space of the GANs to divide the images into different groups for automatic pattern recognition in the dataset. Improving the performance of the clustering techniques that clearly find the difference among clusters, is an area of future work.

GANs and encoder network hybrid models have a good potential to address intra class imbalance problem in face recognition and re-identification tasks. The key idea of these models is to work on latent code space rather than the pixel space. This is because for manipulating fine grained image category, e.g., hair color, the latent code representation will operate only on that single latent code (hair color), whereas the pixel space will edit every single pixel in an image.

The fascinating approaches to use GANs for the problem of object level imbalances in object detection task fall into two general categories: 1. Generating more rare examples as intelligent oversampling used for class imbalance. These generated rare examples are introduced into the training dataset to address imbalance problem. 2. Learn an adversary in combination with original object detection algorithms. This adversary modifies the features to solve imbalance problems instead of generating examples in pixel space. i.e., to generate hard-to-detect samples by performing feature space manipulations.

The capability of super-resolution GANs are being used to up-sample small blurred objects into fine-scale ones and to recover detailed spatial information for accurate small object detection. This technique combines super-resolution GANs with object detection algorithms to solve the imbalances due to object size. The power of adversarial process is being used to increase the diversity of the small object locations in the images by copy-pasting small object instances several times at different locations.

Making the best use of GANs and combining them into U-Net architectures is an interesting way to solve pixel level imbalances in segmentation tasks. These architectures often use weighted loss function to mitigate the pixel level imbalances. Combinations of image inpainting GANs with U-Net architectures has the great potential use in segmenting hidden objects. This technique not only efficient in segmentation tasks, but also to infer appearance of the objects beyond their visible parts. Overall, combining different deep learning models with adversarial process can provide a way to solve many other open problems in computer vision field.

Category	Imbalance Type	Study	Application
Binary Classification	Inter class imbalance	DCGAN [144]	Malaria disease classification
	Inter class imbalance	SDGAN [145]	Industrial defect classification
	Inter class imbalance	BAGAN [146]	Image classification
	Inter class imbalance	CiGAN[147]	Mammogram classification
	Inter class imbalance	CycleGAN[149]	Mammogram classification
	Inter class imbalance	DCGAN [228]	Mammogram classification
	Inter class imbalance	CovidGAN[151]	Covid19 classification
	Intra class imbalance	Clustering + GAN [155]	Imbalanced intra class classification
	Intra class imbalance	Semantically decomposed GAN[156]	Imbalanced intra class classification
	Intra class imbalance	VAE + GAN [107]	Facial Attribute editing
	Intra class imbalance	AttGAN[62]	Facial Attribute editing
	Intra class imbalance	IcGAN[63]	Facial Attribute editing
	Intra class imbalance	ResAttr-GAN [64]	Facial Attribute editing
	Intra class imbalance	ARU-net [163]	Facial Attribute editing
	Intra class imbalance	SaGAN[164]	Facial Attribute editing
	Intra class imbalance	PN-GAN [170]	Person reidentification
	Intra class imbalance	PTGAN [171]	Person reidentification
	Intra class imbalance	CycleGAN[172]	Person reidentification
	Intra class imbalance	SPGAN [173]	Person reidentification

	Intra class imbalance	FDGAN [174]	Person reidentification
	Intra class imbalance	Cross view GAN [176]	Vehicle reidentification
	Intra class imbalance	DCGAN [177]	Vehicle reidentification
	Intra class imbalance	F-CGAN [178]	Fine grained classification
	Intra class imbalance	DCGAN + Fine grained Classifier [182]	Fine grained classification
	Intra class imbalance	General-to-Detailed GAN [184]	Fine grained classification
Multi class Classification	Few minority-many majority class imbalance	Cycle GAN [191]	Emotion classification
	Few minority-many majority class imbalance	DCGAN [192]	Weather classification
	Few minority-many majority class imbalance	DCGAN + Ensemble learning [193]	Weather classification
	Few minority-many majority class imbalance	DCGAN [187]	Chest pathology classification
	Few minority-many majority class imbalance	DCGAN [194]	liver lesion classification
	Many majority- Few minority class imbalance	DCGAN [195]	Skin lesion classification
	Many majority- Many minority class imbalance	Cycle-GAN [186]	Plant disease classification
	Many majority- Many minority class imbalance	WGAN-GP [197]	Multi class classification
	Many majority- Many minority class imbalance	CE-GAN [200]	Multi class classification
Object detection	Object Scale imbalance	Perceptual GAN [203]	Traffic sign detection
	Object Scale imbalance	SOD-MTGAN [208]	Small object detection system
	Object Scale imbalance	Detector GAN [204]	Pedestrian and disease detection
	Imbalance due to occlusions and deformations	Adversarial-Fast-RCNN [211]	Occluded object detection
	Imbalance due to occlusions and deformations	Adversarial Occlusion-aware Face Detector [212]	Occluded face detection
	Imbalance due to occlusions and deformations	Cut-Paste GAN [213]	Occluded object detection
	Foreground Background object class imbalance	Task-aware synthetic data generation [214]	Object detection

	Foreground Background object class imbalance	Gene-GAN [216]	Object detection
	Foreground Background object class imbalance	PSIS [215]	Object detection
	Pixel wise Imbalance	Sensitivity conditional GAN [110]	Shadow detection
	Pixel wise Imbalance	Pix2pix HD GAN [135]	Imbalanced pedestrian image segmentation
	Pixel wise Imbalance	Voxel GAN [221]	Brain tumor segmentation
	Pixel wise Imbalance	GAN + ensemble learning [223]	Medical image semantic segmentation
	Pixel wise Imbalance	GAN + Weighted categorical loss [222]	Heart image segmentation
	Imbalance due to occlusions	SeGAN[226]	Invisible part generation and Segmentation
Segmentation	Imbalance due to occlusions	Occlusion-Aware GAN [227]	Occlusion free image generation

Table 3:Comparative summary of GANs for the problem of imbalances in computer vision

6. Future work

Even though GANs can be used as an effective way to unlock additional information from a dataset, the synthetic images generated by GANs cannot replace the real images completely. However, a blend of different proportions of real and GANs generated images are extremely useful to improve the diversity of the training samples and increase performance of the classifiers. Our future work intends to study the influences of blending different propositions of GANs generated images and real images on the classification performance.

Inflating the size of the dataset brings another problem: One of the most significant limitations in computer vision experiments is computational resource. Sophisticated computer vision models trained on inflated dataset can perform complex tasks, the problem however is, how do we deploy such massive architecture on edge devices for instant usage. Handling this problem using knowledge distillation is non-trivial and an active field of research. Knowledge distillation is model compression technique in which a smaller network is trained with the help of the sophisticated pretrained model to achieving the similar accuracy. This training process is often referred to as "teacher-student", where the sophisticated pretrained model is the teacher and the smaller network is the student. W. Wang et al.[229] combine GANs and knowledge distillation to improve the efficiency of the student network in object detection. Similar to this work, we will attempt to further implement GANs and knowledge distillation combinations to other computer visions tasks.

As research on GANs are developing and maturing, assessment of performance has become essential. Evaluation metrics helps to quantitatively measure how well GANs models are performing, also to assess the relative performance of GANs. Very often the performance of GANs is measured by the manual inspection of the visual fidelity of generated images. However, the manual inspection is cumbersome, subjective, time-consuming, and sometimes misleading. Lack of a universal evaluation metrics can impede the development of GANs. Introducing new performance measure to evaluate both diversity and fidelity of generated images is very important area for future work.

Manually designing GANs architecture for a given task is time-consuming and sometimes have a tendency of errors. This drawback has led researchers to move on to next stage of automating GANs

architecture in the form of neural architecture search (NAS). Another interesting area of further research is to use metaheuristic search algorithms that assist architectural search and find optimal GANs architecture which outperforms human created GANs model.

Achieving equilibrium between the generator and discriminator of the GANs can take a long time relative to other deep neural networks. Distributed training of GAN through parallelization and cluster computing is another important area of future work to cut down the training time.

Most of the applications of the GANs so far have been for creating synthetic images. GANs are not limited to the visual domain and can be also applied to non-visual applications. For example, M. Paganini et al. [230] used GANs to predict the outcome of high energy particle physics experiments. Instead of using explicit Monte Carlo simulation of the real physics of every step, the GANs learns by example what outcome is likely to occur in each situation. The GANs reduces the computational cost of high energy particle simulation, enough to save millions of dollars' worth of supercomputer time. We believe that the invention of new applications using this powerful tool will be continued in the future.

7. Conclusion

This paper surveys various GANs architectures that have been used for addressing the different imbalance problems in computer vision tasks. In this survey, we first provided detailed background information on deep generative models and GAN variants from the architecture, algorithm, and training tricks perspective. In order to present a clear roadmap of various imbalance problems in computer vision tasks, we introduced taxonomy of the imbalance problems. Following the proposed taxonomy, we discussed each type of problems separately in detail and presented the GANs based solutions with important features of each approach and their architectures. We focused mainly on the real-world applications where GAN based synthetic images are used to alleviate class imbalance. In addition to the thorough discussion on the imbalance problems and their solutions, we addressed many open issues that are crucial for computer vision applications.

Synthetic but realistic images generated using the methods discussed in this survey have the potential to mitigate the class imbalance problem while preserving the extrinsic distribution. Many of the methods surveyed in this paper tackled the highly complex imbalances by combining GANs architecture with different other deep learning frameworks. Specifically, the use of autoencoders with GANs has offered an effective way to perform feature space manipulations instead of complex pixel space operations.

Synthetic images generated by GANs cannot be used as the complete replacement for real datasets. However, the blend of real and GANs generated images have enormous potential to increase the performance of the deep learning model. Looking into the future, GAN-related research in image as well as non-image data domains to address the problem of imbalances and limited training dataset would continue to expand. We conclude that the future of GANs is promising and there are clearly a lot of opportunities for further research and applications in many fields.

Abbreviations

Abbreviations	
ConvNets	Convolutional Neural Networks
SMOTE	Synthetic Minority Oversampling Technique
ADASYN	Adaptive synthetic sampling
IHM	Instance hardness measure
SSL	Semi-supervised learning
R-CNN	Region-based Convolutional Neural Networks
RPN	Region Proposal Network
YOLO	You Only Look Once
SSD	Singe Shot Detection
SNIP	Scale Normalization for Image Pyramids
FPN	Feature Pyramid Networks
RNN	Recurrent neural networks
LSTM	Long Short-Term Memory
PCA	Principle component analysis
MADE	Masked Autoencoder Density Estimator
ARs	Autoregressive models
FVBNs	Fully visible belief networks
RGB	Red Green blue
NADE	Neural Autoregressive Density Estimator
MADE	Masked Autoencoder Density Estimator
VAEs	Variational Auto Encoders
CVAE	Conditional Variational Auto Encoders
DC-IGN	Deep Convolutional Inverse Graphics Network
IWVAE	Importance weighted Variational Auto Encoders
VQ-VAEs	Vector Quantized Variational Auto Encoders
DRAW	Deep Recurrent Attentive Writer
EMD	Earth mover Distance
TTUR	Two Time-Scale Update Rule
DDSM	Digital Database for Screening Mammography
ARU-net	Adversarially Regularized U-net
AMN	Attribute manipulation network
SiaNet	Siamese network
CV	Coefficient of variation
AP	Average precision
ASTN	Adversarial Spatial Transformer Network
ASDN	Adversarial Spatial Dropout Network
mAP	Mean Average Precision
AOFD	Adversarial Occlusion Aware Face Detection

PSIS	progressive and selective instance-switching
ADAM	Adaptive Moment Estimation Optimizer
ReLU	Rectified linear unit
GANs	Generative adversarial neural networks
cGAN	conditional Generative Adversarial Network
ACGAN	Auxiliary classifier Generative Adversarial Network
VACGAN	Versatile Auxiliary classifier Generative Adversarial Network
InfoGAN	Information maximizing Generative Adversarial Network
SCGAN	Similarity constraint Generative Adversarial Network
DCGAN	Deep Convolutional Generative Adversarial Network
ProGAN	Progressive Growing of Generative Adversarial Network
LAPGAN	Laplacian Generative Adversarial Network
GRAN	Generative Recurrent Adversarial Networks
D2GAN	Dual discriminator Generative Adversarial Network
MADGAN	Multi-agent diverse Generative Adversarial Network
CoGAN	Coupled Generative Adversarial Network
DEGAN	Decoder Encoder Generative Adversarial Network
VAEGAN	Variational autoencoder Generative Adversarial Network
AAE	Adversarial autoencoders
ALI	Adversarially Learned Inference
BiGAN	Bidirectional Generative Adversarial Network
SRGAN	Super-Resolution Generative Adversarial Network
SAGAN	Self-Attention Generative Adversarial Network
WGAN	Wasserstein Generative Adversarial Network
WGAN-GP	Wasserstein Generative Adversarial Network with gradient penalty
LSGAN	Least squares Generative Adversarial Network
EBGAN	Energy Based Generative Adversarial Network
BEGAN	Boundary Equilibrium Generative Adversarial Network
SD-GAN	surface defect- Generative Adversarial Network
BAGAN	Balancing Generative Adversarial Network
ciGAN	Conditional infilling Generative Adversarial Network
IcGAN	Invertible conditional Generative Adversarial Network
PNGAN	Pose-normalized Generative Adversarial Network
PTGAN	Person Transfer Generative Adversarial Network
SPGAN	Similarity preserving cycle consistent generative adversarial network
FD-GAN	Feature Distilling Generative Adversarial Network
F-CGAN	Fine grained conditional GAN
CE-GAN	Class expert generative adversarial network
ScGAN	Sensitivity conditional Generative Adversarial Network
OAGAN	Occlusion-Aware Generative Adversarial Network

Availability of data and materials

Not applicable

Competing interests:

The authors declare that they have no competing interests.

Funding:

This research work was undertaken in the context of DIGIMAN4.0 project ("Digital Manufacturing Technologies for Zero-defect", https://www.digiman4-0.mek.dtu.dk/). DIGIMAN4.0 is a European Training Network supported by Horizon 2020, the EU Framework Programme for Research and Innovation (Project ID: 814225).

Authors' contributions:

Vignesh Sampath performed the primary literature review and analysis of this survey, and also drafted the manuscript. Iñaki Maurtua, Juan José Aguilar Martín and Aitor Gutierrez worked with Vignesh Sampath to develop the article's framework and focus. Iñaki Maurtua and Juan José Aguilar Martín double checked the manuscript and provided several advanced ideas for this manuscript. All authors read and approved the final manuscript.

Acknowledgements:

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions on the paper. Also we acknowledge the members of the Autonomous and Intelligent Systems Unit, Tekniker, for valuable discussions and collaborations.

References:

[1] B. T. Nugraha, S. F. Su, and Fahmizal, "Towards self-driving car using convolutional neural network and road lane detector," *Proc. 2nd Int. Conf. Autom. Cogn. Sci. Opt. Micro Electro-Mechanical Syst. Inf. Technol. ICACOMIT 2017*, vol. 2018-Janua, pp. 65–69, 2017, doi: 10.1109/ICACOMIT.2017.8253388.

[2] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0276-2.

[3] A. Gutierrez, A. Ansuategi, L. Susperregi, C. Tubío, I. Rankić, and L. Lenža, "A Benchmarking of Learning Strategies for Pest Detection and Identification on Tomato Plants for Autonomous Scouting Robots Using Internal Databases," *J. Sensors*, vol. 2019, 2019, doi: 10.1155/2019/5219471.

[4] L. Santos, F. N. Santos, P. M. Oliveira, and P. Shinde, "Deep Learning Applications in Agriculture: A Short Review," 2020, pp. 139–151.

[5] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, "A fast and robust convolutional neural network-based defect detection model in product quality control," *Int. J. Adv. Manuf. Technol.*, vol. 94, no. 9–12, pp. 3465–3471, 2018, doi: 10.1007/s00170-017-0882-0.

[6] M. Hashemi, "Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0263-7.

[7] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 2, pp. 1097–1105, 2012.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[10] C. Szegedy *et al.*, “Going Deeper with Convolutions,” *CoRR*, vol. abs/1409.4, Sep. 2014.
[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *CVPR*, 2015.
[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.
[14] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.
[15] S. Al-Stouhi and C. K. Reddy, “Transfer learning for class imbalance problems with inadequate data,” *Knowl. Inf. Syst.*, vol. 48, no. 1, pp. 201–228, Jul. 2016, doi: 10.1007/s10115-015-0870-3.
[16] A. Ali, S. M. Shamsuddin, and A. L. Ralescu, “Classification with class imbalance problem: A review,” *Int. J. Adv. Soft Comput. its Appl.*, vol. 7, no. 3, pp. 176–204, 2015.
[17] J. Zhang, Y. Xia, Q. Wu, and Y. Xie, “Classification of Medical Images and Illustrations in the Biomedical Literature Using Synergic Deep Learning,” no. June 2017, 2017.
[18] Q. Dong, S. Gong, and X. Zhu, “Imbalanced Deep Learning by Minority Class Incremental Rectification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1367–1381, Jun. 2019, doi: 10.1109/TPAMI.2018.2832629.
[19] Y. Zhang, B. Li, H. Lu, A. Irie, and X. Ruan, “Sample-Specific SVM Learning for Person Re-identification,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1278–1287, doi: 10.1109/CVPR.2016.143.
[20] M. M. Sawant and K. M. Bhurchandi, “Age invariant face recognition: a survey on facial aging databases, techniques and effect of aging,” *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 981–1008, Aug. 2019, doi: 10.1007/s10462-018-9661-z.
[21] E. Mostafa, A. Ali, N. Alajlan, and A. Farag, “Pose Invariant Approach for Face Recognition at Distance,” Springer, Berlin, Heidelberg, 2012, pp. 15–28.
[22] N. Japkowicz and S. Stephen, “The class imbalance problem: A systematic study,” *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Nov. 2002, doi: 10.3233/IDA-2002-6504.
[23] N. V. Chawla, “Data Mining for Imbalanced Datasets: An Overview,” in *Data Mining and Knowledge Discovery Handbook*, New York: Springer-Verlag, pp. 853–867.
[24] N. V. Chawla, N. Japkowicz, and A. Kotcz, “Special Issue on Learning from Imbalanced Data Sets,” *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 1–6, Jun. 2004, doi: 10.1145/1007730.1007733.
[25] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” Jun. 2011, doi: 10.1613/jair.953.
[26] Haibo He, Yang Bai, E. A. Garcia, and Shutao Li, “ADASYN: Adaptive synthetic sampling approach for imbalanced learning,” in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1322–1328, doi: 10.1109/IJCNN.2008.4633969.
[27] K. PUNTUMAPON, T. RAKTHAMAMON, and K. WAIYAMAI, “Cluster-Based Minority Over-Sampling for Imbalanced Datasets,” *IEICE Trans. Inf. Syst.*, vol. E99.D, no. 12, pp. 3101–3109, 2016, doi: 10.1587/transinf.2016EDP7130.
[28] M. R. Smith, T. Martinez, and C. Giraud-Carrier, “An instance level analysis of data complexity,” *Mach. Learn.*, vol. 95, no. 2, pp. 225–256, May 2014, doi: 10.1007/s10994-013-5422-z.
[29] P. Y. Simard, D. Steinkraus, and J. C. Platt, “Best practices for convolutional neural networks applied to visual document analysis,” in *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, vol. 1, pp. 958–963, doi: 10.1109/ICDAR.2003.1227801.
[30] J. Lemley, S. Bazrafkan, and P. Corcoran, “Deep Learning for Consumer Devices and Services: Pushing the limits for machine learning, artificial intelligence, and computer vision.,” *IEEE Consum. Electron. Mag.*, vol. 6, no. 2, pp. 48–56, Apr. 2017, doi: 10.1109/MCE.2016.2640698.
[31] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.
[32] H. Wu and S. Prasad, “Semi-Supervised Deep Learning Using Pseudo Labels for Hyperspectral Image Classification,” *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1259–1270, Mar. 2018, doi: 10.1109/TIP.2017.2772836.
[33] J. E. van Engelen and H. H. Hoos, “A survey on semi-supervised learning,” *Mach. Learn.*, vol. 109, no. 2,

pp. 373–440, Feb. 2020, doi: 10.1007/s10994-019-05855-6.
[34] N. Thai-Nghe, Z. Gantner, and L. Schmidt-Thieme, "Cost-sensitive learning methods for imbalanced data," in *The 2010 International Joint Conference on Neural Networks (IJCNN)*, 2010, pp. 1–8, doi: 10.1109/IJCNN.2010.5596486.
[35] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.
[36] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.
[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.
[38] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020, doi: 10.1109/TPAMI.2018.2844175.
[39] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," 2016, pp. 21–37.
[40] J. S. D. R. G. A. F. Redmon, "(YOLO) You Only Look Once," *Cvpr*, 2016, doi: 10.1109/CVPR.2016.91.
[41] B. Singh and L. S. Davis, "An Analysis of Scale Invariance in Object Detection - SNIP," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3578–3587, doi: 10.1109/CVPR.2018.00377.
[42] F. Yang, W. Choi, and Y. Lin, "Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2129–2137, doi: 10.1109/CVPR.2016.234.
[43] B. Singh, M. Najibi, and L. S. Davis, "SNIPER: Efficient Multi-Scale Training," May 2018.
[44] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 936–944, doi: 10.1109/CVPR.2017.106.
[45] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
[46] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 304–311, doi: 10.1109/CVPR.2009.5206631.
[47] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random Erasing Data Augmentation," Aug. 2017.
[48] X. Wang, A. Shrivastava, and A. Gupta, "A-Fast-RCNN: Hard Positive Generation via Adversary for Object Detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3039–3048, doi: 10.1109/CVPR.2017.324.
[49] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440, doi: 10.1109/CVPR.2015.7298965.
[50] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.
[51] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," 2015, pp. 234–241.
[52] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data," *ISPRS J. Photogramm. Remote Sens.*, vol. 162, pp. 94–114, Apr. 2020, doi: 10.1016/j.isprsjprs.2020.01.013.
[53] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," Jun. 2019, doi: 10.1109/ACCESS.2020.2983149.
[54] D. Tabernik, S. Šela, J. Skvarč, and D. Skočaj, "Segmentation-Based Deep-Learning Approach for Surface-Defect Detection," Mar. 2019, doi: 10.1007/s10845-019-01476-x.
[55] I. Rizwan I Haque and J. Neubert, "Deep learning approaches to biomedical image segmentation," *Informatics Med. Unlocked*, vol. 18, p. 100297, 2020, doi: 10.1016/j.imu.2020.100297.
[56] M. Cordts *et al.*, "The Cityscapes Dataset for Semantic Urban Scene Understanding," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 3213–3223, 2016, doi: 10.1109/CVPR.2016.350.

[57] B. H. Menze *et al.*, “The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS),” *IEEE Trans. Med. Imaging*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015, doi: 10.1109/TMI.2014.2377694.
[58] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation,” in *2016 Fourth International Conference on 3D Vision (3DV)*, 2016, pp. 565–571, doi: 10.1109/3DV.2016.79.
[59] W. R. Crum, O. Camara, and D. L. G. Hill, “Generalized Overlap Measures for Evaluation and Validation in Medical Image Analysis,” *IEEE Trans. Med. Imaging*, vol. 25, no. 11, pp. 1451–1461, Nov. 2006, doi: 10.1109/TMI.2006.880587.
[60] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, “Tversky Loss Function for Image Segmentation Using 3D Fully Convolutional Deep Networks,” 2017, pp. 379–387.
[61] M. Berman, A. R. Triki, and M. B. Blaschko, “The Lovasz-Softmax Loss: A Tractable Surrogate for the Optimization of the Intersection-Over-Union Measure in Neural Networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4413–4421, doi: 10.1109/CVPR.2018.00464.
[62] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, “AttGAN: Facial Attribute Editing by Only Changing What You Want,” *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5464–5478, Nov. 2019, doi: 10.1109/TIP.2019.2916751.
[63] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, “Invertible Conditional GANs for image editing,” Nov. 2016.
[64] R. Tao, Z. Li, R. Tao, and B. Li, “ResAttr-GAN: Unpaired Deep Residual Attributes Learning for Multi-Domain Face Image Translation,” *IEEE Access*, vol. 7, pp. 132594–132608, 2019, doi: 10.1109/ACCESS.2019.2941272.
[65] I. J. Goodfellow *et al.*, “Generative adversarial nets,” *Adv. Neural Inf. Process. Syst.*, vol. 3, no. January, pp. 2672–2680, 2014, doi: 10.3156/jsoft.29.5_177_2.
[66] C. Bowles *et al.*, “GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks,” Oct. 2018.
[67] M. I. J. T. J. Sejnowski, “Learning and Relearning in Boltzmann Machines,” *Graph. Model. Found. Neural Comput. MITP*, 2001.
[68] D. E. R. J. L. McClelland, “Information Processing in Dynamical Systems: Foundations of Harmony Theory,” *Parallel Distrib. Process. Explor. Microstruct. Cogn. Found. , MITP*, pp. 194–281, 1987.
[69] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science (80-.).*, vol. 313, no. 5786, pp. 504–507, 2006, doi: 10.1126/science.1127647.
[70] R. Salakhutdinov and G. Hinton, “Deep Boltzmann machines,” *J. Mach. Learn. Res.*, vol. 5, no. 3, pp. 448–455, 2009.
[71] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, “Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations,” *Comput. Sci. Dep. Stanford Univ.*, p. 8, 2009.
[72] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006, doi: 10.1162/neco.2006.18.7.1527.
[73] P. Ramachandran *et al.*, “Fast Generation for Convolutional Autoregressive Models,” Apr. 2017.
[74] B. J. Frey, *Graphical models for machine learning and digital communication*. MIT Press, 1998.
[75] B. J. Frey, G. E. Hinton, and P. Dayan, “Does the Wake-sleep Algorithm Produce Good Density Estimators?,” *Adv. Neural Inf. Process. Syst.*, vol. 13, no. 1, pp. 661–670, 1996.
[76] B. Uria, M.-A. Côté, K. Gregor, I. Murray, and H. Larochelle, “Neural Autoregressive Distribution Estimation,” May 2016.
[77] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[78] L. Theis and M. Bethge, “Generative Image Modeling Using Spatial LSTMs,” Jun. 2015.
[79] A. Krizhevsky, “Learning multiple layers of features from tiny images,” 2009.
[80] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.
[81] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, “Pixel Recurrent Neural Networks,” Jan. 2016.
[82] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, “Conditional Image Generation with PixelCNN Decoders,” Jun. 2016.
[83] T. Salimans, A. Karpathy, X. Chen, and D. P. Kingma, “PixelCNN++: Improving the PixelCNN with Discretized Logistic Mixture Likelihood and Other Modifications,” Jan. 2017.
[84] X. Chen, N. Mishra, M. Rohaninejad, and P. Abbeel, “PixelSNAIL: An Improved Autoregressive Generative Model,” Dec. 2017.

[85] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th international conference on Machine learning - ICML ’08*, 2008, vol. 311, pp. 1096–1103, doi: 10.1145/1390156.1390294.
[86] P. Baldi, *Autoencoders, Unsupervised Learning, and Deep Architectures*. PMLR, 2012.
[87] A. Y. Ng, “Sparse autoencoder.”
[88] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction,” 2011, pp. 52–59.
[89] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, “Contractive Auto-Encoders: Explicit Invariance During Feature Extraction,” in *ICML*, 2011.
[90] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” Dec. 2013.
[91] S. Tan and B. Li, “Stacked convolutional auto-encoders for steganalysis of digital images,” in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, 2014, pp. 1–4, doi: 10.1109/APSIPA.2014.7041565.
[92] M. Germain, K. Gregor, I. Murray, and H. Larochelle, “MADE: Masked Autoencoder for Distribution Estimation,” Feb. 2015.
[93] K. Sohn, X. Yan, and H. Lee, “Learning structured output representation using deep conditional generative models,” *Adv. Neural Inf. Process. Syst.*, vol. 2015-Janua, pp. 3483–3491, 2015.
[94] I. Higgins *et al.*, “B-VAE: Learning basic visual concepts with a constrained variational framework,” *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, pp. 1–13, 2019.
[95] T. D. Kulkarni, W. Whitney, P. Kohli, and J. B. Tenenbaum, “Deep Convolutional Inverse Graphics Network,” Mar. 2015.
[96] Y. Burda, R. Grosse, and R. Salakhutdinov, “Importance Weighted Autoencoders,” Sep. 2015.
[97] I. Gulrajani *et al.*, “PixelVAE: A Latent Variable Model for Natural Images,” Nov. 2016.
[98] X. Chen *et al.*, “Variational Lossy Autoencoder,” Nov. 2016.
[99] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, “DRAW: A Recurrent Neural Network For Image Generation,” Feb. 2015.
[100] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, “Neural Discrete Representation Learning,” Nov. 2017.
[101] A. Razavi, A. van den Oord, and O. Vinyals, “Generating Diverse High-Fidelity Images with VQ-VAE-2,” Jun. 2019.
[102] F. Huszár, “How (not) to Train your Generative Model: Scheduled Sampling, Likelihood, Adversary?,” Nov. 2015.
[103] W. Lotter, G. Kreiman, and D. Cox, “Deep Predictive Coding Networks for Video Prediction and Unsupervised Learning,” May 2016.
[104] A. Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” Nov. 2015.
[105] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, “Adversarial Autoencoders,” Nov. 2015.
[106] V. Dumoulin *et al.*, “Adversarially Learned Inference,” Jun. 2016.
[107] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” Dec. 2015.
[108] G. Zhong, W. Gao, Y. Liu, and Y. Yang, “Generative Adversarial Networks with Decoder-Encoder Output Noise,” Jul. 2018.
[109] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton, “VEEGAN: Reducing Mode Collapse in GANs using Implicit Variational Learning,” May 2017.
[110] M. Mirza and S. Osindero, “Conditional Generative Adversarial Nets,” Nov. 2014.
[111] A. Odena, C. Olah, and J. Shlens, “Conditional Image Synthesis With Auxiliary Classifier GANs,” Oct. 2016.
[112] S. Bazrafkan and P. Corcoran, “Versatile Auxiliary Classifier with Generative Adversarial Network (VAC+GAN), Multi Class Scenarios,” Jun. 2018.
[113] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, “InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets,” Jun. 2016.
[114] X. Li, L. Chen, L. Wang, P. Wu, and W. Tong, “SCGAN: Disentangled Representation Learning by Adding Similarity Constraint on Generative Adversarial Nets,” *IEEE Access*, vol. 7, pp. 147928–147938, 2019, doi: 10.1109/ACCESS.2018.2872695.
[115] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” Jan. 2017.
[116] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved Training of Wasserstein GANs,” Mar. 2017.

[117] H. Petzka, A. Fischer, and D. Lukovnicov, “On the regularization of Wasserstein GANs,” Sep. 2017.
[118] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, “Least Squares Generative Adversarial Networks,” Nov. 2016.
[119] J. Zhao, M. Mathieu, and Y. LeCun, “Energy-based Generative Adversarial Network,” Sep. 2016.
[120] D. Berthelot, T. Schumm, and L. Metz, “BEGAN: Boundary Equilibrium Generative Adversarial Networks,” Mar. 2017.
[121] R. Wang, A. Cully, H. J. Chang, and Y. Demiris, “MAGAN: Margin Adaptation for Generative Adversarial Networks,” Apr. 2017.
[122] J. Zhao *et al.*, “Dual-agent GANs for photorealistic and identity preserving profile face synthesis,” *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. 15, pp. 66–76, 2017.
[123] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation,” Oct. 2017.
[124] E. Denton, S. Chintala, A. Szlam, and R. Fergus, “Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks,” Jun. 2015.
[125] D. J. Im, C. D. Kim, H. Jiang, and R. Memisevic, “Generating images with recurrent adversarial networks,” Feb. 2016.
[126] T. D. Nguyen, T. Le, H. Vu, and D. Phung, “Dual Discriminator Generative Adversarial Nets,” Sep. 2017.
[127] A. Ghosh, V. Kulharia, V. Namboodiri, P. H. S. Torr, and P. K. Dokania, “Multi-Agent Diverse Generative Adversarial Networks,” Apr. 2017.
[128] M.-Y. Liu and O. Tuzel, “Coupled Generative Adversarial Networks,” Jun. 2016.
[129] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, “Learning to Discover Cross-Domain Relations with Generative Adversarial Networks,” Mar. 2017.
[130] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251, doi: 10.1109/ICCV.2017.244.
[131] C. Ledig *et al.*, “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network,” Sep. 2016.
[132] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Sep. 2014.
[133] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-Attention Generative Adversarial Networks,” May 2018.
[134] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976, doi: 10.1109/CVPR.2017.632.
[135] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, “High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8798–8807, doi: 10.1109/CVPR.2018.00917.
[136] M. G. Bellemare *et al.*, “The Cramer Distance as a Solution to Biased Wasserstein Gradients,” May 2017.
[137] Y. Mroueh, T. Sercu, and V. Goel, “McGan: Mean and Covariance Feature Matching GAN,” Feb. 2017.
[138] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Póczos, “MMD GAN: Towards Deeper Understanding of Moment Matching Network,” May 2017.
[139] Y. Mroueh and T. Sercu, “Fisher GAN,” May 2017.
[140] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved Techniques for Training GANs,” Jun. 2016.
[141] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, “Amortised MAP Inference for Image Super-resolution,” Oct. 2016.
[142] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium,” Jun. 2017.
[143] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, “Spectral Normalization for Generative Adversarial Networks,” Feb. 2018.
[144] L. M. Shoohi and J. H. Saud, “Dcgan for handling imbalanced malaria dataset based on over-sampling technique and using cnn,” *Medico-Legal Updat.*, vol. 20, no. 1, 2020, doi: 10.3750/v20/i1/2020/mlu/194444.
[145] S. Niu, B. Li, X. Wang, and H. Lin, “Defect Image Sample Generation With GAN for Improving Defect Recognition,” *IEEE Trans. Autom. Sci. Eng.*, pp. 1–12, 2020, doi: 10.1109/TASE.2020.2967415.
[146] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, and C. Malossi, “BAGAN: Data Augmentation with Balancing GAN,” Mar. 2018.

[147] E. Wu, K. Wu, D. Cox, and W. Lotter, “Conditional Infilling GANs for Data Augmentation in Mammogram Classification,” 2018, pp. 98–106.
[148] M. Heath, K. Bowyer, D. Kopans, R. Moore, and W. P. Kegelmeyer, “Digital Database for Screening Mammography.” [Online]. Available: https://www.mammoimage.org/databases/.
[149] C. Muramatsu *et al.*, “Improving breast mass classification by shared data with domain transformation using a generative adversarial network,” *Comput. Biol. Med.*, vol. 119, p. 103698, Apr. 2020, doi: 10.1016/j.compbiomed.2020.103698.
[150] S. Guan, “Breast cancer detection using synthetic mammograms from generative adversarial networks in convolutional neural networks,” *J. Med. Imaging*, vol. 6, no. 03, p. 1, Mar. 2019, doi: 10.1117/1.JMI.6.3.031411.
[151] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, “CovidGAN: Data Augmentation Using Auxiliary Classifier GAN for Improved Covid-19 Detection,” *IEEE Access*, vol. 8, pp. 91916–91923, 2020, doi: 10.1109/ACCESS.2020.2994762.
[152] “COVID-19 Chest X-Ray Dataset Initiative.” [Online]. Available: https://github.com/agchung/Figure1-COVID-chestxray-dataset.
[153] “IEEE Covid Chest X-Ray Dataset.” [Online]. Available: https://github.com/ieee8023/covid-chestxray-dataset.
[154] “Covid19 Radiography Database.”
[155] N. Hase, S. Ito, N. Kanaeko, and K. Sumi, “Data augmentation for intra-class imbalance with generative adversarial network,” in *Fourteenth International Conference on Quality Control by Artificial Vision*, 2019, p. 56, doi: 10.1117/12.2521692.
[156] C. Donahue, Z. C. Lipton, A. Balsubramani, and J. McAuley, “Semantically Decomposing the Latent Spaces of Generative Adversarial Networks,” May 2017.
[157] Y. Wang *et al.*, “Orthogonal Deep Features Decomposition for Age-Invariant Face Recognition,” 2018, pp. 764–779.
[158] D. Gong, Z. Li, D. Lin, J. Liu, and X. Tang, “Hidden Factor Analysis for Age Invariant Face Recognition,” in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 2872–2879, doi: 10.1109/ICCV.2013.357.
[159] X. Yin and X. Liu, “Multi-Task Convolutional Neural Network for Pose-Invariant Face Recognition,” *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 964–975, Feb. 2018, doi: 10.1109/TIP.2017.2765830.
[160] P. Carcagnì, M. Del Coco, D. Cazzato, M. Leo, and C. Distante, “A study on different experimental configurations for age, race, and gender estimation problems,” *EURASIP J. Image Video Process.*, vol. 2015, no. 1, p. 37, Dec. 2015, doi: 10.1186/s13640-015-0089-y.
[161] L. Ziwei, L. Ping, W. Xiaogang, and X. Tang, “Large-scale CelebFaces Attributes (CelebA) Dataset,” 2018. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html.
[162] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, “AttGAN: Facial Attribute Editing by Only Changing What You Want,” *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5464–5478, Nov. 2019, doi: 10.1109/TIP.2019.2916751.
[163] J. Zhang, A. Li, Y. Liu, and M. Wang, “Adversarially Regularized U-Net-based GANs for Facial Attribute Modification and Generation,” *IEEE Access*, vol. 7, pp. 86453–86462, 2019, doi: 10.1109/ACCESS.2019.2926633.
[164] G. Zhang, M. Kan, S. Shan, and X. Chen, “Generative Adversarial Network with Spatial Attention for Face Attribute Editing,” 2018, pp. 422–437.
[165] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, “Invertible Conditional GANs for image editing,” Nov. 2016.
[166] Z. Zheng, X. Yang, Z. Yu, L. Zheng, Y. Yang, and J. Kautz, “Joint Discriminative and Generative Learning for Person Re-Identification,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2133–2142, doi: 10.1109/CVPR.2019.00224.
[167] X. Zhang and Y. Gao, “Face recognition across pose: A review,” *Pattern Recognit.*, vol. 42, no. 11, pp. 2876–2896, Nov. 2009, doi: 10.1016/j.patcog.2009.04.017.
[168] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, “Face recognition from a single image per person: A survey,” *Pattern Recognit.*, vol. 39, no. 9, pp. 1725–1745, Sep. 2006, doi: 10.1016/j.patcog.2006.03.013.
[169] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition,” *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399–458, Dec. 2003, doi: 10.1145/954339.954342.
[170] X. Qian *et al.*, “Pose-Normalized Image Generation for Person Re-identification,” 2018, pp. 661–678.
[171] L. Wei, S. Zhang, W. Gao, and Q. Tian, “Person Transfer GAN to Bridge Domain Gap for Person Re-

identification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 79–88, doi: 10.1109/CVPR.2018.00016.
[172] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang, "Camera Style Adaptation for Person Re-identification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5157–5166, doi: 10.1109/CVPR.2018.00541.
[173] W. Deng, L. Zheng, Q. Ye, Y. Yang, and J. Jiao, "Similarity-preserving Image-image Domain Adaptation for Person Re-identification," Nov. 2018.
[174] Y. Ge *et al.*, "FD-GAN: Pose-guided Feature Distilling GAN for Robust Person Re-identification," *Adv. Neural Inf. Process. Syst.*, vol. 2018-Decem, no. NeurIPS, pp. 1222–1233, 2018.
[175] A. Zheng, X. Lin, C. Li, R. He, and J. Tang, "Attributes Guided Feature Learning for Vehicle Re-identification," May 2019.
[176] Y. Zhou and L. Shao, "Cross-View GAN Based Vehicle Generation for Re-identification," in *Procedings of the British Machine Vision Conference 2017*, 2017, doi: 10.5244/C.31.186.
[177] F. Wu, S. Yan, J. S. Smith, and B. Zhang, "Vehicle re-identification in still images: Application of semi-supervised learning and re-ranking," *Signal Process. Image Commun.*, vol. 76, pp. 261–271, Aug. 2019, doi: 10.1016/j.image.2019.04.021.
[178] Y. Fu, X. Li, and Y. Ye, "A multi-task learning model with adversarial data augmentation for classification of fine-grained images," *Neurocomputing*, vol. 377, pp. 122–129, Feb. 2020, doi: 10.1016/j.neucom.2019.10.002.
[179] Z. Ge, A. Bewley, C. McCool, P. Corke, B. Upcroft, and C. Sanderson, "Fine-grained classification via mixture of deep convolutional neural networks," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–6, doi: 10.1109/WACV.2016.7477700.
[180] A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, "Novel dataset for fine-grained image categorization," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011.
[181] P. Welinder, S. Branson, T. Mita, C. Wah, and F. Schroff, "Caltech-ucsd Birds 200," *Caltech-UCSD Tech. Rep.*, vol. 200, pp. 1–15, 2010, doi: CNS-TR-2010-001.
[182] C. Wang, Z. Yu, H. Zheng, N. Wang, and B. Zheng, "CGAN-plankton: Towards large-scale imbalanced class generation and fine-grained classification," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 855–859, doi: 10.1109/ICIP.2017.8296402.
[183] E. C. Orenstein, O. Beijbom, E. E. Peacock, and H. M. Sosik, "WHOI-Plankton- A Large Scale Fine Grained Visual Recognition Benchmark Dataset for Plankton Classification," 2015.
[184] T. Koga, N. Nonaka, J. Sakuma, and J. Seita, "General-to-Detailed GAN for Infrequent Class Medical Images," Nov. 2018.
[185] X. Zhu, Y. Liu, Z. Qin, and J. Li, "Data Augmentation in Emotion Classification Using Generative Adversarial Networks," Nov. 2017.
[186] D. S. P. Haseeb Nazki, Jaehwan Lee, Sook Yoon, "Image-to-Image Translation with GAN for Synthetic Data Augmentation in Plant Disease Datasets," *Smart Media J.*, vol. 8, no. 22019, pp. 46–57, 2019, doi: DOI: 10.30693/SMJ.2019.8.2.46.
[187] H. Salehinejad, S. Valaee, T. Dowdell, E. Colak, and J. Barfett, "Generalization of Deep Neural Networks for Chest Pathology Classification in X-Rays Using Generative Adversarial Networks," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 2018-April, pp. 990–994, 2018, doi: 10.1109/ICASSP.2018.8461430.
[188] Y.-W. Lu, K.-L. Liu, and C.-Y. Hsu, "Conditional Generative Adversarial Network for Defect Classification with Class Imbalance," in *2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*, 2019, pp. 146–149, doi: 10.1109/SMILE45626.2019.8965320.
[189] Shuo Wang and Xin Yao, "Multiclass Imbalance Problems: Analysis and Potential Solutions," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 42, no. 4, pp. 1119–1130, Aug. 2012, doi: 10.1109/TSMCB.2012.2187280.
[190] Shuo Wang and Xin Yao, "Multiclass Imbalance Problems: Analysis and Potential Solutions," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 42, no. 4, pp. 1119–1130, Aug. 2012, doi: 10.1109/TSMCB.2012.2187280.
[191] X. Zhu, Y. Liu, Z. Qin, and J. Li, "Data Augmentation in Emotion Classification Using Generative Adversarial Networks," Nov. 2017.
[192] Z. Li, Y. Jin, Y. Li, Z. Lin, and S. Wang, "Imbalanced Adversarial Learning for Weather Image Generation and Classification," in *2018 14th IEEE International Conference on Signal Processing (ICSP)*, 2018, pp. 1093–1097, doi: 10.1109/ICSP.2018.8652272.
[193] Y. Huang, Y. Jin, Y. Li, and Z. Lin, "Towards Imbalanced Image Classification: A Generative Adversarial Network Ensemble Learning Method," *IEEE Access*, vol. 8, pp. 88399–88409, 2020, doi:

10.1109/ACCESS.2020.2992683.
[194] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, “GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification,” *Neurocomputing*, vol. 321, pp. 321–331, Dec. 2018, doi: 10.1016/j.neucom.2018.09.013.
[195] H. Rashid, M. A. Tanveer, and H. Aqeel Khan, “Skin Lesion Classification Using GAN based Data Augmentation,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 916–919, doi: 10.1109/EMBC.2019.8857905.
[196] P. Tschandl, C. Rosendahl, and H. Kittler, “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” *Sci. Data*, vol. 5, no. 1, p. 180161, Dec. 2018, doi: 10.1038/sdata.2018.161.
[197] S. Bhatia and R. Dahyot, “Using WGAN for Improving Imbalanced Classification Performance,” in *AICS 2019*, 2019.
[198] A. Krizhevsky, “Learning multiple layers of features from tiny images. Tech. rep., CIFAR-10 (Canadian Institute for Advanced Research),” 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html.
[199] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms,” pp. 1–6, 2017.
[200] Fanny and T. W. Cenggoro, “Deep Learning for Imbalance Data Classification using Class Expert Generative Adversarial Network,” *Procedia Comput. Sci.*, vol. 135, pp. 60–67, 2018, doi: 10.1016/j.procs.2018.08.150.
[201] T. Y. Lin *et al.*, “Microsoft COCO: Common objects in context,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8693 LNCS, no. PART 5, pp. 740–755, 2014, doi: 10.1007/978-3-319-10602-1_48.
[202] H. Bai, S. Wen, and S. H. G. Chan, “Crowd counting on images with scale variation and isolated clusters,” *Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019*, pp. 18–27, 2019, doi: 10.1109/ICCVW.2019.00009.
[203] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, “Perceptual Generative Adversarial Networks for Small Object Detection,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1951–1959, doi: 10.1109/CVPR.2017.211.
[204] L. Liu, M. Muelly, J. Deng, T. Pfister, and L. J. Li, “Generative modeling for small-data object detection,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 6072–6080, 2019, doi: 10.1109/ICCV.2019.00617.
[205] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, “Traffic-Sign Detection and Classification in the Wild,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2110–2118, doi: 10.1109/CVPR.2016.232.
[206] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The Pascal Visual Object Classes (VOC) Challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.
[207] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian Detection: An Evaluation of the State of the Art,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012, doi: 10.1109/TPAMI.2011.155.
[208] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, “SOD-MTGAN: Small object detection via multi-task generative adversarial network,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11217 LNCS, no. i, pp. 210–226, 2018, doi: 10.1007/978-3-030-01261-8_13.
[209] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.
[210] S. C. B, N. Koznek, A. Ismail, G. Adam, V. Narayan, and M. Schulze, *Computer Vision – ECCV 2018 Workshops*, vol. 11133, no. March. 2019.
[211] X. Wang, A. Shrivastava, and A. Gupta, “A-Fast-RCNN: Hard Positive Generation via Adversary for Object Detection,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3039–3048, doi: 10.1109/CVPR.2017.324.
[212] Y. Chen, L. Song, and R. He, “Adversarial Occlusion-aware Face Detection,” Sep. 2017, doi: 1709.05188.
[213] D. Dwibedi, I. Misra, and M. Hebert, “Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1310–1319, doi: 10.1109/ICCV.2017.146.
[214] S. Tripathi, S. Chandra, A. Agrawal, A. Tyagi, J. M. Rehg, and V. Chari, “Learning to Generate Synthetic Data via Compositing,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*

(CVPR), 2019, pp. 461–470, doi: 10.1109/CVPR.2019.00055.
[215] H. Wang, Q. Wang, F. Yang, W. Zhang, and W. Zuo, "Data Augmentation for Object Detection via Progressive and Selective Instance-Switching," Jun. 2019.
[216] S. Zhou, T. Xiao, Y. Yang, D. Feng, Q. He, and W. He, "GeneGAN: Learning Object Transfiguration and Object Subspace from Unpaired Data," in *Procedings of the British Machine Vision Conference 2017*, 2017, doi: 10.5244/C.31.111.
[217] S. Liu, J. Zhang, Y. Chen, Y. Liu, Z. Qin, and T. Wan, "Pixel Level Data Augmentation for Semantic Image Segmentation Using Generative Adversarial Networks," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 1902–1906, doi: 10.1109/ICASSP.2019.8683590.
[218] V. Nguyen, T. F. Y. Vicente, M. Zhao, M. Hoai, and D. Samaras, "Shadow Detection with Conditional Generative Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4520–4528, doi: 10.1109/ICCV.2017.483.
[219] J. Zhu, K. G. G. Samuel, S. Z. Masood, and M. F. Tappen, "Learning to recognize shadows in monochromatic natural images," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 223–230, doi: 10.1109/CVPR.2010.5540209.
[220] T. F. Y. Vicente, L. Hou, C.-P. Yu, M. Hoai, and D. Samaras, "Large-Scale Training of Shadow Detectors with Noisily-Annotated Shadow Examples," 2016, pp. 816–832.
[221] M. Rezaei, H. Yang, and C. Meinel, "voxel-GAN: Adversarial Framework for Learning Imbalanced Brain Tumor Segmentation," 2019, pp. 321–333.
[222] M. Rezaei, H. Yang, and C. Meinel, "Recurrent generative adversarial network for learning imbalanced medical image semantic segmentation," *Multimed. Tools Appl.*, vol. 79, no. 21–22, pp. 15329–15348, Jun. 2020, doi: 10.1007/s11042-019-7305-1.
[223] M. Rezaei, H. Yang, and C. Meinel, "Conditional Generative Refinement Adversarial Networks for Unbalanced Medical Image Semantic Segmentation," Oct. 2018.
[224] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, and K. Lewis, "Sensors and systems for fruit detection and localization: A review," *Comput. Electron. Agric.*, vol. 116, pp. 8–19, Aug. 2015, doi: 10.1016/j.compag.2015.05.021.
[225] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A Fruit Detection System Using Deep Neural Networks," *Sensors*, vol. 16, no. 8, p. 1222, Aug. 2016, doi: 10.3390/s16081222.
[226] K. Ehsani, R. Mottaghi, and A. Farhadi, "SeGAN: Segmenting and Generating the Invisible," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6144–6153, doi: 10.1109/CVPR.2018.00643.
[227] J. Dong, L. Zhang, H. Zhang, and W. Liu, "Occlusion-Aware GAN for Face De-Occlusion in the Wild," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6, doi: 10.1109/ICME46284.2020.9102788.
[228] S. Guan, "Breast cancer detection using synthetic mammograms from generative adversarial networks in convolutional neural networks," *J. Med. Imaging*, vol. 6, no. 03, p. 1, Mar. 2019, doi: 10.1117/1.JMI.6.3.031411.
[229] W. Wang, W. Hong, F. Wang, and J. Yu, "GAN-Knowledge Distillation for One-Stage Object Detection," *IEEE Access*, vol. 8, pp. 60719–60727, 2020, doi: 10.1109/ACCESS.2020.2983174.
[230] M. Paganini, L. de Oliveira, and B. Nachman, "CaloGAN: Simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks," *Phys. Rev. D*, vol. 97, no. 1, p. 014021, Jan. 2018, doi: 10.1103/PhysRevD.97.014021.