Autoencoders and Variational Autoencoders

Hanin Monir Ismail 192895
Artificial intelligence major
ICS, The British University in Egypt
Cairo, Egypt
hanin192895@bue.edu.eg

Jacinta Samir 206562
Artificial intelligence major
ICS, The British University in Egypt
Cairo, Egypt
jacinta206562@bue.edu.eg

Youssef Ayman 203800
Artificial intelligence major
ICS, The British University in Egypt
Cairo, Egypt
youssef203800@bue.edu.eg

***Abstract* – Autoencoders are a type of neural network that encode their input in a lower dimensional latent space, and decode it, using the most meaningful information derived, to reconstruct the input as closely as possible. Many differing architectures and types of autoencoders have been developed across various applications, the most prominent of which are surveyed in this literature review. Following the survey, the models proposed in the papers are compared and critiqued, finding that hybrid autoencoder networks typically outperform their simpler counterparts.**

Keywords – Encoder, Decoder, Autoencoder, Variational Autoencoder, Denoising, Anomaly Detection, Dynamic Rendering, Photo-geometric Reconstruction, Audio Synthesis

I. Introduction

Autoencoders (AEs) are unsupervised neural networks that learn identity functions to be able to reconstruct the input data [1]. They compress the high-dimensional input data to be able to learn sparse representations and reconstruct the input from bare essentials. This can be useful for many applications, such as denoising. They consist of encoders, which reduce the high- dimensional input into low dimensional latent code, and decoders, which recover the data from the latent space. The architecture seen in "Fig. 1" forms a bottleneck, as the encoder layer sizes decrease, and decoder layer sizes increase, such that the input and output layers consist of the same number of neurons [2].

The properties of AEs allow them to be utilized in a field of various applications, across an abundance of domains. The nature of the bottleneck architecture makes AEs useful for dimensionality reduction and image compression, and the latent space representation of input allows for feature extraction and subsequently, anomaly detection. Building upon this, since only the most pivotal features are represented in the latent space and are used for reconstruction, this makes AEs a powerful tool for image denoising.

A wide spectrum of network architectures has been built based off the foundations of autoencoders. Most notably, Variational Autoencoders (VAEs). VAEs build on the concept of reconstruction by being able to generate new images after learning the sparse representations [3]. They do so by describing the latent space in a probabilistic manner, such that the relationship between input data and its encoding vector can be defined by prior, likelihood, and posterior probabilities. In VAEs, the encoder layers can be referred to as the recognition model, and the decoder referred to as the generative model.

In this paper, the state-of-the-art in AEs and their variations will be discussed and critiqued, whilst describing the architecture of each model. Following the literature review, the models and their characteristics will be compared, across assorted evaluation metrics.

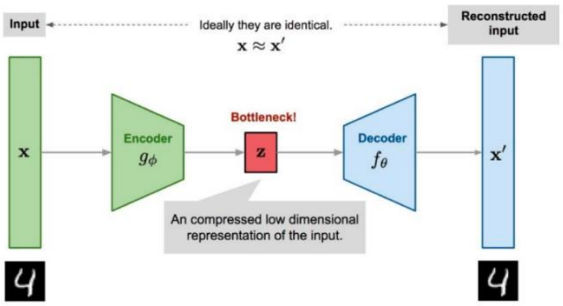

Fig. 1. Architecture an autoencoder [2]

II. Literature Review

Autoencoders and variational autoencoders have been widely used in different applications, such as image denoising, anomaly detection, image reconstruction and many others. In this section we discuss and analyze previous works done using AEs and VAEs in various fields, distinctly noting their architectures and characteristics.

A. Autoencoders Based Deep Learner for Image Denoising

Bajaj et al. [4] proposes the use of an autoencoder-based deep learning model to remove Gaussian noise from images. The proposed model uses convolutional layers, deconvolutional layers, and linearly connected convolutional denoising autoencoder (CDA) blocks to learn the degradation from training images and remove it from the input image [4]. The performance of the model is evaluated using peak signal to noise ratio (PSNR) and Structural Similarity Index (SSIM) [4]. The architecture of the network consists of an input layer, convolution layer, deconvolution layer and output layer [4]. In the convolutional layer, several small and linearly connected convolutional denoising autoencoder (CDA) blocks are used [4]. CDA works by learning from trained images and tries to enhance the input image [4]. CDA consists of four internal layers which are: Convolution layer, Pooling layer, Deconvolution layer and Up-sampling layer [4]. The activation function that is used in this model is Parametric Rectified Linear Unit (PRELU). In addition to this, batch normalization is placed between the convolution and the pooling layers, and again between the deconvolution and the up-sampling layers [4]. The dataset used is STL-10, which contains 100000 unlabeled images, each 96*96 pixels in size [4]. The proposed denoising network used an unsupervised learning-based model, with 40000 unlabeled images used to train the model [4]. The testing was applied on the set5 dataset, which has a set of five standard images [4]. Standard deviations of the added gaussian noise were 10, 30, 50, and 70 [4]. The degraded original image is then restored using the proposed model [4]. The peak signal-to-noise ratio (PSNR) compares the original and reconstructed

images, with a high PSNR value indicating great image quality [4]. SSIM measures the differences between the original image and the reconstructed image [4]. If the value equals 1 then the two images are identical, and if the value is less than 1 then the two images are different [4]. The model performs better than CDNN in terms of PSNR [4]. RED30 performs well when the standard deviation is high [4]. The value of SSIM can be reduced in limited cases [4]. They conclude that it is reasonable to assume that their suggested model outperforms CDNN and RED30 with respect to PSNR values [4].

B. *Dual Autoencoder Network for Retinex-Based Low-Light Image Enhancement*

In continuance to the application of AEs to image enhancement, Park et al. [5] introduce a model based on the retinex theory, which consists of a dual autoencoder network, used to perform low-light enhancement and noise reduction on images [5]. The model combines stacked and convolutional autoencoders [5]. The model estimates the enhanced image in three stages: firstly, it estimates the illumination component through a stacked autoencoder, then it provides an initial estimation of the reflectance component, and finally, it refines the reflectance component by utilizing a convolutional autoencoder [5]. The method proposed reduces the amplified noise in the enhanced reflectance component through the use of a convolutional autoencoder [5]. While a conventional denoising autoencoder trains on randomly corrupted input and original vectors to reduce dimensionality, a convolutional autoencoder trains on two-dimensional image data and can better preserve image structure by sharing weights across all input data [5]. In the proposed method, the convolutional autoencoder acts as a constraint term on the reflectance component through L1-norm minimization [5]. Furthermore, the proposed dual autoencoder is trained to provide output equal to input in a self-supervised learning manner, and the stacked and convolutional autoencoders serve as the data-fidelity term in the retinex model [5]. Because it is challenging to find naturally occurring low- and high-contrast image pairs, they created a set of training data by generating a pair of low- and high-contrast patches [5]. To synthesize the low-contrast image patch, they sampled approximately 80,000 patches of size 33×33 from a set of ideal, high-contrast images gathered from the internet and a dataset used in a different paper [5]. To train the convolutional denoising autoencoder, they needed a large amount of data consisting of pairs of noisy and noise-free image patches [5]. Since it is difficult to obtain naturally occurring pairs with varying levels of noise, they synthesized their own, using the luminance channel of the HSV color space from about 190,000 ideal image patches [5]. The noisy patches were created by adding Gaussian noise with randomly selected standard deviations between 10 and 18, while the noise-free patches were simply the original images [5]. The resulting pairs of patches were all 28x28 pixels in size [5]. The denoising performance of the convolutional autoencoder was evaluated using the mean squared error (MSE) metric at various numbers of convolution filters and receptive field sizes of the max-pooling layer [5]. It was observed that using a smaller size for the receptive field in the max-pooling layer and increasing the number of convolution filters resulted in improved denoising performance with a smaller MSE value in both training and validation [5]. As a result, the proposed method employs a local window of size 2×2 to take the maximum value [5]. Moreover, their proposed method is tested using an objective assessment that evaluates the PSNR and the SSIM [5]. The retinex-based dual autoencoder model proposed is able to enhance images more effectively than current image enhancement methods, as it can produce superior results without the issues of saturation and amplified noise [5].

C. *Unsupervised Anomaly Video Detection via a Double-Flow ConvLSTM Variational Autoencoder*

Wang et al. [6] propose a VAE model to detect video anomalies using unsupervised data. However, because of the generalization limitations of VAEs and their inability to consider the temporal dependence of data, (thus cannot process time-series data) they combine VAE with both a convolutional network and a long-term short-term memory (LSTM) to construct their models [6]. They construct 2 models, using ConvLSTM [6] The first model, “ConvLSTM-VAE”, is an asymmetric model created by weakening the decoder [6]. The model consists of 3 main components an encoder, a sample, and a decoder [6]. The encoder has 2 main parts the “Conv”, which is a set of convolutional layers that extracts the spatial features from each video frame, and “ConvLSTM”, which learns the temporal patterns from the spatial features extracted [6]. Next, in the sample phase, a data point z is obtained from the encoder with temporal and spatial properties [6]. Finally, the decoder consists of a single “Deconv” module, which is a set of deconvolutional layers corresponding to the encoder's Conv module [6]. The “Deconv” module generates new realistic inputs based on the z data point. The second model they introduce is an improvement to the first model, called DF-ConvLSTM-VAE [6]. The DF-ConvLSTM-VAE model learns temporal patterns in video sequences [6]. This model’s architecture is based on 2 flows, a left flow and a right flow [6]. The right flow of the network is same as the structure of the ConvLSTM-VAE, while the left flow is different. It consists of three parts: encoder, sampler, and decoder. Unlike the ConvLSTM-VAE(Asymmetric) model, the encoder has two modules: Conv and ConvLSTM of the right flow [6]. The sampler has two sample processes, one for the right flow data z sampled from $N(\mu,\sigma 2)$ and the other for the left flow data z' sampled from $N(\mu',\sigma'2)$ [6]. Finally, the decoder module is Deconv. Their constructed network models the probabilistic distribution of the data from a normal video in an unsupervised scheme and reconstructs videos, removing any anomalies, in an attempt to detect video anomalies [6]. They use the reconstruction error probability (REP) to calculate the anomaly score [6]. First, they calculate the REP of a pixel intensity value of a specific location in a frame of a video, then sum up all the calculated pixel error within a frame to estimate the frame’s REP [6]. Then they compute a regularity score for a video sequence using the calculated frame REPs [6]. To determine the count of anomaly incidents in a video, they investigate the noisy and insignificant local minima in the time-series of the regularity scores [6]. Different local minima signify the likelihood of the presence of anomalies in specific video frames [6]. To identify significant local minima, they employ the Persistence1D algorithm [6]. During this step, if the distance between two local minima is less than 50 frames, it is recognized as components of the same abnormal incident [6]. They used the USCD dataset, which is divided in to 2 parts, ped1 and ped2, and the Avenue dataset, each frame was

resized and converted to grayscale [6]. Furthermore, they evaluated their model using ROC curve, AUC, and equal error rate (EER), which are commonly used with video anomaly detection models [6]. They train a VAE, 2 ConvLSTM-VAE, one symmetric and 1 unsymmetric, and their DF-ConvLSTM-VAE and compare them based on their achieved AUC, EER, and computation time [6]. On ped1 of the USCD dataset, the symmetric model of the ConvLSTM-VAE achieved the highest AUC, 89.4, and the lowest EER, 16.4%, while on ped2, the ConvLSTM-VAE also achieved the highest AUC, 88.9, and the EER decreased to 14.1% [6]. However, the DF-ConvLSTM-VAE scored the lowest EER on ped2, being 12.2% [6]. In addition, when trained on the Avenue dataset, the DF-ConvLSTM-VAE model scored the highest AUC, 87.2, while the symmetric ConvLSTM-VAE scored the lowest EER, being 18.5% [6]. The results indicate that the DF-ConvLSTM-VAE model outperforms the ConvLSTM-VAE(Asymmetric) model [6].

D. Unsupervised Learning of Probably Symmetric Deformable 3D Objects from Images in the Wild

In an unconventional approach to the applications of autoencoders, Wu et al. [7] built a photo geometric autoencoder to create 3D reconstructions from 2D single view images without a 3D ground truth [7]. The autoencoder decomposes the single view 2D images into 4 factors, lighting, depth, albedo (the amount of light reflected by a surface), and viewpoint [7]. The components of the objects are separated in an unsupervised way, taking advantage of the fact that many object categories have a symmetric structure [7]. To achieve this, they utilize illumination information to exploit the underlying symmetry of objects, even when their appearance is not symmetrical due to shading [7]. Additionally, they predict a symmetry probability map for objects that are likely to be symmetric, which is learned alongside the other parts of the model [7]. The given image I is a function that takes values from the set Ω and maps them to the set R^3 [7]. Ω represents a grid defined by the range $\{0, \ldots, W - 1\} \times \{0, \ldots, H - 1\}$, or alternatively, a tensor in $R^3 \times W \times H$ [7]. The image is assumed to be approximately centered on an object of interest [7]. The objective is to learn a function Φ, implemented as a neural network, which takes the image I as input and outputs four factors - a depth map $d: \Omega \rightarrow R^+$, an albedo image $a: \Omega \rightarrow R^3$, a global light direction $l \in S^2$, and a viewpoint $w \in R^6$ [7]. These factors are used to reconstruct the image [7]. Their method is tested on multiple datasets, 3 human face datasets, CelebA dataset, 3DFAW, and BFM. CelebA is a dataset of human faces with over 200,000 real images annotated with bounding boxes, while 3DFAW has 23,000 images with 66 3D key point annotations, used for evaluating 3D predictions [7]. The images are roughly cropped around the head region, and the official train/Val/test splits are used [7]. The synthetic face model BFM (Basel Face Model) is used to assess the quality of 3D reconstructions since in-the-wild datasets lack ground truth [7]. The method is also test on 2 cat images datasets [7]. After the model generates a depth map d in the canonical view, they transform it to a depth map $\bar{d}$ in the actual view using the predicted viewpoint [7]. Then compare the transformed depth map with the ground-truth depth map d* using the scale-invariant depth error (SIDE) [7]. Along with SIDE, the degree to which the surface is captured is measured by the mean angle deviation (MAD) between normals computed from the predicted depth and those computed from the ground truth depth [7]. To further examine the performance of their model, they also perform an ablation study to examine the influence of each part of the model [7]. The depth and albedo are created using encoder-decoder networks, while the viewpoint and lighting are estimated using simpler encoder networks [7]. Because the output is in the canonical viewpoint, skip connections are not used in the encoder-decoders, as the input and output images are not spatially aligned [7]. The same network is used to predict all four confidence maps, which are computed at different resolutions for the photometric and perceptual losses. The final activation function for depth, albedo, viewpoint, and lighting is tanh, while it is SoftPlus for the confidence maps [7]. Before tanh is applied to the depth prediction, it is centered on the mean as the viewpoint includes estimating the global distance [7]. They conclude that their current model performs better than a leading 3D reconstruction method that relies on 2D key point supervision but is limited to some cases like images with harsh lighting, noisy texture or extreme side poses [7]. Additionally, the model only represents 3D shapes using a depth map from a canonical viewpoint, which is suitable for objects with a roughly convex shape and a clear canonical viewpoint, such as faces [7].

E. Neural Volumes: Learning Dynamic Renderable Volumes from Images

Following in Wu et al. [7] ‘s steps, Lombardi et al. [8] utilize an autoencoder to create a 3D volume rendering from 2D images. The technique comprises two primary parts: an encoder-decoder neural network that converts input images into a 3D volume V(x), and a ray-marching step that is differentiable and renders an image from the volume V with camera parameters [8]. The approach is thought of as an autoencoder where the final layer performs a fixed-function, volume rendering operation without any trainable parameters [8] The method involves capturing a set of Multiview captures, and then using an encoder network to produce a latent code z that represents the state of the scene at each time instant, based on a subset of the camera images [8]. A volume decoder network is then used to generate a 3D volume V (x; z) for each latent code, which assigns an RGBα value to each point x in the volume [[8]. Finally, an accumulative ray-marching algorithm renders the volume from a specific viewpoint [8]. A lot of the memory is allocated to modeling the inside of objects or empty space, which does not affect the rendered output [8]. This is due to the regular grid structure of the representation [8]. To address this issue, they use a warping technique that allows the learning algorithm to better utilize the available memory [8]. By doing this, they achieve higher fidelity compared to using only a traditional voxel data structure [8]. The entire system is trained end-to-end by reconstructing each input image and minimizing the squared pixel reconstruction loss across the entire training set, with the complete pipeline being run during training to optimize the encoder-decoder network weights [8]. The encoder architecture consists of separate branches for each camera view that are then combined and further encoded to produce the final representation [8]. To create the latent space, the state of the scene at a particular moment is represented by encoding a subset of views from a multi-camera system using a convolutional neural network (CNN) [8]. They argue that to be able to produce believable samples through the latent space, the generative model needs to be able to effectively extrapolate from the training data [8]. To achieve that, they use a variational architecture to smoothen the latent space [8]. To ensure that a given radiance value maps to the same pixel value for each camera, they introduce a gain (д) and bias (b)

for each camera and channel [8]. These are applied to the reconstructed image before comparing it to the ground truth [8]. By doing this, the model is able to account for any small variations in overall intensity in the image [8]. To restrict the algorithm to reconstruct only the object are of interest, they estimate a separate background image I(bg)_rgb for each camera [8]. This background image remains static throughout the entire sequence and captures only stationary objects that are typically outside of the reconstruction volume [8]. Because, without reconstruction priors, the reconstructed volumes may contain artifacts that resemble smoke, due to variations in appearance from different angles caused by view-dependent effects or calibration errors [8]. To address this issue, they introduce a small amount of opacity in a particular camera to allow the system to learn how to compensate for such differences [8]. To mitigate these artifacts, two priors are introduced [8]. The first prior imposes regularization on the total variation of the log voxel opacities, while the second prior is a beta distribution that is applied to the final image opacities with parameters (0.5, 0.5) [8]. The model is trained on 3 dataset a moving hand, swinging hair, and dry ice smoke [8]. They use the Adam optimizer to optimize the loss function [8]. The model is evaluated using multiple qualitative and quantitative aspects, including MSE [8]. The MSE scores indicate that the model is more capable of representing the scene when viewed from different angles [8]. They use single frames to test the quality of their model [8]. Their experiments prove that their method is capable of modeling complex phenomena such as fuzz, smoke, and human skin and hair, as demonstrated by the renderings [8]. However, the reconstructed images may contain some artifacts, such as a light smokey pattern, which is likely caused by view-dependent appearance for certain training cameras, but adding more camera views can help reduce these artifacts [8]. Their model has the capability to represent non-glossy specular highlights by considering the viewpoint, but it cannot accurately represent highly detailed specular highlights with high frequency [8].

F. "RAVE: A variational autoencoder for fast and high-quality neural audio synthesis

Caillon et al [9] introduce a Realtime Audio Variational autoencoder (RAVE) for fast and high-quality audio waveform synthesis. RAVE uses a two-stage training procedure, representation learning, and adversarial fine-tuning, and a post-training analysis of the latent space for control between reconstruction fidelity and representation compactness [9]. The first stage is representation learning, where the model is trained to learn a latent space representation that captures important perceptual features about the audio signal [9]. The second stage is adversarial fine-tuning, where the model is trained to generate synthesized signals like the real ones by relying on a discriminator [9]. The paper also proposes a method for identifying the most informative parts of the latent space to restrict the dimensionality of the learned representation to the minimum required to reconstruct a signal [9]. This is done by applying a Singular Value Decomposition (SVD) on a matrix of latent space samples and removing the variance from collapsed dimensions [9]. The remaining dimensions are informative and are used to represent the audio signal [9]. The audio synthesis model consists of an encoder and a modified decoder [9]. The encoder combines multiband decomposition and a convolutional neural network to convert the raw audio waveform into a 128-dimensional latent representation [9]. The modified decoder uses upsampling layers, residual networks, and three sub-networks to synthesize a multiband audio signal with a tanh activation, which is multiplied by an amplitude envelope generated by a loudness sub-network with a sigmoid activation, which is a strided convolutional network applied on different scales of the audio signal to prevent artifacts [9]. The model is trained for 3M steps with the Adam optimizer, dequantization, random crop, and allpass filters with random coefficients as data augmentation strategies [9]. The proposed model aims to reduce artifacts in silent parts and increase the naturalness of noisy signals in audio synthesis [9]. RAVE leverages multi-band decomposition of the raw waveform and can generate 48kHz audio signals, while running 20 times faster than real-time on a standard laptop CPU [9]. Their paper evaluates synthesis quality using quantitative and qualitative subjective experiments and demonstrates its superiority over existing models [9]. Applications of the model include timbre transfer and signal compression [9]. Generative models are used to model the underlying distribution of a given dataset by introducing latent variables that account for variations in the data [9]. However, the joint distribution of these variables and the data is often too complex to solve analytically [9]. Variational autoencoders solve this problem by introducing an inference model that approximates the true posterior distribution of the latent variables given the data [9]. The Evidence Lower Bound (ELBO) is used to train the model by minimizing the reconstruction error of the data and regularizing the posterior distribution to match a predefined prior [9]. The posterior and prior distributions are parametrized by the encoder and the decoder, respectively [9]. By weighting the KL divergence with a parameter beta, the trade-off between reconstruction accuracy and latent regularization can be controlled [9]. Increasing beta leads to less entangled latent dimensions but lower reconstruction quality [9]. The raw waveform modelling task can be addressed using models like WaveNet and SampleRNN, which factorize the probability of a waveform as a product of conditional probabilities [9]. However, these models require a large amount of data and parameters to converge properly, and the autoregressive nature of the synthesis process makes it slow and prone to errors [9].

They conclude that RAVE sounds better than NSynth and SING without autoregressive generation [9]. It also synthesizes high-quality audio with 3.5 times less parameters, however, ground truth and other theories still differ in evaluation [9]. The synthesis speed of audio generation is measured as the average number of samples produced per second in 100 trials [9]. NSynth is the slowest model due to its autoregressive synthesis, peaking at 57Hz during generation [9]. On the other hand, SING and RAVE are much faster due to their parallel nature [9]. RAVE, which incorporates multiband decomposition, is 25 times faster than the original and outperforms SING on both CPU and GPU [9].

G. Unsupervised Outlier Detection via Transformation Invariant Autoencoder

Cheng et al propose a novel unsupervised outlier detection method called Transformation Invariant Autoencoder (TIAE)

for complex image datasets [10]. The proposed method uses a transformation invariant autoencoder that incorporates adaptive self-paced learning to select the most confident inliers for training, helping to mitigate the negative effects of noise introduced by outliers and stabilize network training [10]. They evaluate TIAEs performance against existing autoencoder-based methods on five image datasets and show that TIAE outperforms other methods by up to 10% AUROC [10]. The proposed method combines two techniques: transformation invariant representation learning and self-paced learning [10]. Transformation invariant representation learning aims to learn representations that are consistent under transformations, while self-paced learning simulates the process of human learning by starting with easier aspects of a task and gradually moving towards more complex ones [10]. The authors use image transformations as a means of data augmentation and select transformations that erase specific information that is much shared among inliers and little shared among outliers [10]. The TIAE framework is based on an encoder-decoder architecture, where the encoder captures high-level features, and the decoder restores the original image from transformed images [10]. The authors use 2 losses for training and 1 loss for testing to measure the distance between restored images and targets [10]. They also incorporate adaptive self-paced learning to select the most confident examples gradually to mitigate the negative effect of outliers [10]. The study evaluates TIAE's performance on five public datasets: MNIST, Fashion-MNIST, SVHN, CIFAR-10, and CIFAR-100 [10]. They construct an image set with outliers where inliers are images of one class with the same semantic concept, and outliers are randomly sampled from the rest of the classes by an outlier ratio [10]. The performance of TIAE is compared with existing state-of-the-art autoencoder-based UOD methods [10]. The evaluation metrics used are AUROC and AUPR, and experiments are repeated five times to report average results [10]. Additionally, they conducted an ablation study to analyze the contributions of two parts of the proposed TIAE framework: transformation invariant autoencoder and self-paced learning module [10]. The study concludes that TIAE is effective for unsupervised outlier detection and works much better on complex datasets like CIFAR-10 compared to conventional autoencoder-based methods [10]. The proposed method can be extended to other deep unsupervised outlier detection algorithms and has potential for applications in anomaly detection and data cleaning [10].

III. Comparative Analysis

The aforementioned papers are compared and evaluated in this section, with regards to their architectural properties and overall performance.

A. Denoising and Low-Light Enhancement

Bajaj et al. [4] and Park et al. [5] both proposed the use of AEs for image correction, in particular denoising and low-light enhancement respectively. The evaluation metrics used were the following:

- Peak Signal-to-Noise Ratio (PSNR): measures the highest signal-to-noise ratio between the original image and the reconstructed image. A high PSNR indicates good quality of reconstruction.
- Structural Similarity Index (SSIM): measures the perceptual difference between an image and its reconstruction. An SSIM of 1 means that the reconstruction and the original image are exactly the same.

The values of PSNR and SSIM for both models can be seen in Table I. It can be seen that Bajaj et al.'s autoencoder network outperformed with regards to both performance metrics. However, the two models were implemented on differing datasets, and were made to accomplish slightly different results, since low-light enhancement is a more specified form of denoising. For such reasons a direct comparison of their performance is not befitting.

The architecture also varies between the two. Park et al. implemented two AES. First, a stacked AE for illumination estimation, which follows traditional AE architecture but with multiple hidden layers. Second, a convolutional AE for reflectance estimation, which utilized a single convolutional layer in the encoder and two in the decoder. Both models implement convolutional (and pooling layers), as Bajaj et al.'s model comprised multiple CDA blocks (as previously described). Bajaj et al.'s model overall consisted of a convolutional layer, ten CDA blocks (each four layers deep), and two deconvolutional layers before the final output layer. This makes Bajaj et al.'s model significantly deeper than its counterpart. Though Bajaj et al. did not comment on the model's running time, it can be deduced that their model would likely require greater compute resources and be more time intensive.

Since the deeper model, utilizing repeated CDA blocks, was able to more closely replicate and reconstruct its input images, perhaps applying a similar architecture to the low-light problem may improve results. This is especially since low-light noise may be more a complex feature to pinpoint (than the normally distributed Gaussian noise removed by Bajaj et al.), so a more sophisticated network would likely achieve better results.

B. Anomaly Detection

Of the reviewed studies, Wang et al. [6] and Cheng et al. [10] introduced autoencoder-based networks to perform anomaly detection, in both videos and images respectively. Each network was applied on multiple datasets. Table II depicts the Area Under Curve (AUC) scores for each model on each dataset used, as well as an overall average score per model. From this, it is evident that the DF-Conv-LSTM-VAE model proposed by Wang et al. was able to achieve better results across the board, having a significantly higher average score. In all five datasets utilized by Cheng et al.'s TIAE the AUC was lower than for the three datasets used for the DF-Conv-LSTM-VAE. Needless to say, the datasets were different for both models, so a direct comparison cannot be made.

TABLE I. Evaluation scores for two autoencoder denoising models: Bajaj et al. [4] and Park et al. [5].

Paper	**Evaluation Metric**	
	PSNR	***SSIM***
Bajaj et al.'s AE	27.61	0.806
Park et al.'s dual AE	17.02	0.704

TABLE II. VALUES FOR THE AREA UNDER ROC FOR TWO OUTLIER DETECTION AUTOENCODER MODELS: WANG ET AL. [6] AND CHENG ET AL. [10].

Paper	AUC (%)	
	On Each Dataset	*Average*
Wang et al.'s DF-Conv-LSTM-VAE	88.4; 88.8; 87.2[a]	88.13
Cheng et al.'s TIAE	85.18; 86.77; 68.78; 71.18; 65.01[b]	75.28

a. On Ped1, Ped2 and Authentic datasets respectively
b. On MNIST, F-MNIST, SVHN, CIFAR-10 and CIFAR-100 datasets respectively.

The differing nature of the datasets, i.e. video vs image, does not play a pivotal role in the outcomes of the models, as any video is processed frame-by-frame, acting as a series of images. For this reason, the TIAE can be easily applied to a dataset of videos rather than images (and vice-versa). Thus, the DF-Conv-LSTM-VAE seems as though it would likely outperform the TIAE on its 5 datasets, as well as others. This is because the TIAE was able to achieve better scores on simpler datasets such as MNIST, and perform remarkably worse on more complicated data, like CIFAR, whereas the datasets performed well in by the DF-Conv-LSTM-VAE involved more intricate data than MNIST.

Be that as it may, the DF-Conv-LSTM-VAE still displayed certain weaknesses. For instance, if a target object was small, details were harder to derive, and perhaps the object was mistakenly thought to be anomalous. This was because the background of the frame was more likely to distract the model from the foreground, and in such cases would primarily need to be removed. The network was also built based on Gaussian distribution assumptions, which makes it unable to decently encompass real-world complexity.

C. 3D Reconstruction

Wu et al. [7] and Lombardi et al. [8] presented autoencoder-based models that generated 3D reconstructions of objects from their two-dimensional representations. Wu et al. produced 3D reconstructions from single views, whereas Cheng et al. exploited multiple views of an object. The two studies assessed the performance of their models using different evaluation metrics, and thus a score-based comparison cannot be made. To attempt to comparatively analyze their results, a qualitative approach will be taken.

"Fig. 2" and "Fig. 3" show sample outputs for both models, Wu et al. and Cheng et al. respectively. Both models were able to not only reconstruct their input, but also form 3D models which can then be repositioned and adapted. Although Wu et al.'s model was able to produce a satisfactory 3D reconstruct from simply a single image, there were failures it experienced, as seen in "Fig. 4." These were namely in cases where the input image depicted extreme lighting variations, extreme poses, or noisy texture.

Even in the ideal scenarios depicted by "Fig. 2," it is sometimes apparent that the reconstruction is slightly distorted and of lower quality (especially when re-lighting). This suggests that a single view of an object is not adequate for a comprehensive 3D reconstruction. On the other hand, the multi-view approach taken by Cheng et al. produced much more advanced and detailed 3D reconstructions. Moreover, it was able to do so on more challenging objects that portray movement and intricacy, such as hair and smoke. Cheng et al. were able to content a novel model that attained impressive

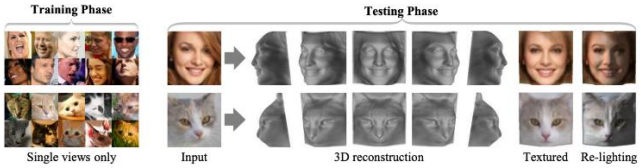

Fig 2: Unsupervised learning of 3D deformable images from in-the-wild images, proposed by Wu et al. [7]

Fig. 3. Qualitative results of Lombardi et al.'s model, showing ground truth, reconstructed image, and visualization of RMSE for each pixel, for 3 datasets of 3 viewpoints each [8].

Fig. 4. Failure cases of Wu et al.'s proposed model [7].

resolution, thus being able to overcome one of the most inherent limitations of autoencoders: blurry reconstructions.

D. Audio Synthesis

Caillon et al.'s [9] Realtime Audio Variational autoencoder (RAVE) was evaluated via a qualitative experiment, wherein 33 participants (most of which being audio professionals) were asked to rate a plethora of audio samples on a scale from 1 to 5. Alongside the audio clips produced by RAVE, the participants queried samples reconstructed by two other competing models, as well as the ground truth (from the Strings dataset). The authors found that the Mean Opinion Score (MOS) for their RAVE model was 3.01, with the ground truth audios achieving the highest score of 4.21. This was the highest MOS received amongst the three models assessed.

It was further demonstrated that the RAVE model was able to synthesize real-time audio 20 times faster on a laptop CPU, with the introduction of multiband decomposition (even reaching 240 times faster with a GPU). Applying this multiband decomposition on the raw waveform to down-sample the temporal dimensionality of the data has proven to be a valuable addition to the application of audio synthesis, as it reduces time spent during both training and synthesis, whilst still maintaining high quality output.

IV. CONCLUSION

In this paper, several state-of-the-art neural networks based on the principles of autoencoders have been surveyed, inspected, and analyzed. It was seen that autoencoders have inspired the development of an extensive array of varying models, across numerous domains and diversified data types. In particular, autoencoder architecture has shown its strength when combined with other neural network designs, such as

convolutional layers and Long Short-Term Memory networks (LSTMs). This interweaving and evolving of network architectures not only tailors the model to the specific needs and facets of an application, but also overcomes the weaknesses of a simpler autoencoder. Needless to say, there then becomes a tradeoff between complexity and computation, so a suitable balance needs to be found. Autoencoders have already proved to be helpful tools for certain tasks, such as denoising, but in the future, they can continue to be applied to generative tasks. New variants can be developed, incorporating autoencoders with elements of other neural networks, to attain more true-to-life reconstructions. Many of the reviewed papers provide open-source implementations of their models, giving opportunities for improvement.

REFERENCES

[1] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science (1979)*, vol. 313, no. 5786, 2006, doi: 10.1126/science.1127647.

[2] L. Weng, "From autoencoder to beta-vae," *Lil'Log (Alt + H)*, Aug. 12, 2018. https://lilianweng.github.io/posts/2018-08-12-vae/. (Accessed Mar. 05, 2023).

[3] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends in Machine Learning*, vol. 12, no. 4. 2019. doi: 10.1561/2200000056.

[4] K. Bajaj, D. K. Singh, and M. A. Ansari, "Autoencoders Based Deep Learner for Image Denoising," *Procedia Comput Sci*, vol. 171, pp. 1535–1541, Jan. 2020, doi: 10.1016/J.PROCS.2020.04.164.

[5] S. Park, S. Yu, M. Kim, K. Park, and J. Paik, "Dual Autoencoder Network for Retinex-Based Low-Light Image Enhancement," *IEEE Access*, vol. 6, pp. 22084–22093, 2018, doi: 10.1109/ACCESS.2018.2812809.

[6] L. Wang, H. Tan, F. Zhou, W. Zuo, and P. Sun, "Unsupervised Anomaly Video Detection via a Double-Flow ConvLSTM Variational Autoencoder," *IEEE Access*, vol. 10, pp. 44278–44289, 2022, doi: 10.1109/ACCESS.2022.3165977.

[7] S. Wu, C. Rupprecht, and A. Vedaldi, "Unsupervised Learning of Probably Symmetric Deformable 3D Objects from Images in the Wild," Nov. 2019.

[8] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh, "Neural Volumes: Learning Dynamic Renderable Volumes from Images," *ACM Trans Graph*, vol. 38, no. 4, pp. 1–14, Aug. 2019, doi: 10.1145/3306346.3323020.

[9] A. Caillon and P. Esling, "RAVE: A variational autoencoder for fast and high-quality neural audio synthesis," Nov. 2021.

[10] Z. Cheng, E. Zhu, S. Wang, P. Zhang, and W. Li, "Unsupervised Outlier Detection via Transformation Invariant Autoencoder," *IEEE Access*, vol. 9, pp. 43991–44002, 2021, doi: 10.1109/ACCESS.2021.3065838.