IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 5, 2020, accepted August 16, 2020, date of publication August 20, 2020, date of current version September 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3018228

Conditional Introspective Variational Autoencoder for Image Synthesis

KUN ZHENG[1], YAFAN CHENG[2], XIAOJUN KANG[2], (Member, IEEE), HONG YAO[2], (Member, IEEE), AND TIAN TIAN[2], (Member, IEEE)

[1]School of Geography and Information Engineering, China University of Geosciences, Wuhan 430074, China
[2]School of Computer Science, China University of Geosciences, Wuhan 430074, China

Corresponding author: Tian Tian (tiantian@cug.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1004600, in part by the National Science and Technology Major Project of China under Grant 2017ZX05036-001-010, in part by the Science and Technology Planning Project of Guangdong Province, China, under Grant 2018B020207012, and in part by the National Natural Science Foundation of China under Grant 41701417, Grant 61972365, and Grant 61672474.

ABSTRACT We present a variational autoencoder (VAE) learning framework with introspective training for conditional image synthesis, and explore conditional capsule encoder by class-wise mask label insertion for this framework. Our model only consists of encoder (E), generator (G) and classifier (C), where E and G can be adversarially optimized, and C helps to boost conditional generation, improve authenticity and provide generation measures for E and G. Discriminator is not necessary in our framework and its absence makes our model more concise with fewer artifacts and pattern collapse problems. To compensate for the blurry weakness of VAE-like models, feature matching is introduced into loss functions by means of C to offer more reasonable measures between real and synthesized images. Moreover, in consideration of the key role of E in autoencoders as well as the interesting characteristics of capsule structure, conditional capsule encoder is preliminary explored in the image synthesis model. Class labels participate conditional encoding by masking high-level capsules of other categories, and capsule loss for the encoder is added to facilitate conditional synthesis. Experiments on MNIST and Fashion-MNIST data sets show that our model achieves real conditional synthesis performances with better diversity and fewer artifacts. And conditional capsule encoder also reveals interesting synthesis effects.

INDEX TERMS Image generation, artificial neural networks, image processing.

I. INTRODUCTION

Image synthesis is an important issue in many visual tasks. To build generative models for image synthesis, distribution of images should be learned or simulated. Then by adding some random ingredients in the generation process, the models aim to produce realistic but different images from the real ones [1]–[4]. This task is considered as very difficult in computer vision fields, since the distribution of sample images may lie on a high dimensional manifold which is hard to represent with simple models and low dimension parameters [5], [6]. Conditional image synthesis is even more challenging for the sake of condition control. The conditional generative model must be able to not only produce synthetic images, but also generate images of a given category [7].

Owing to the development of deep learning, conditional image synthesis has achieved great progress by designing deep convolutional neural networks [5], [6], [8], [9]. Among these deep generative models, the Conditional Variational Autoencoder (CVAE) [10] and the Conditional Generative Adversarial Network (CGAN) [11] are the most typical and popular models which lead two genres respectively. CVAE derives from the autoencoder, which encodes the latent variables based on conditional probability distributions. It is able to generate images with the architecture of an encoder and a generator, however the synthetic images are easy to be blurry since it is hard to estimate the difference between the generated image and the real image. CGAN realizes conditional synthesis on the basis of Generative Adversarial Networks

The associate editor coordinating the review of this manuscript and approving it for publication was Siddhartha Bhattacharyya.

(GANs) [2], where both generator and discriminator are fed by extra class labels and finally achieve Nash equilibrium by adversarial training. CGAN is able to produce clearer images, whereas the capability limit of the discriminator and the game between encoder and discriminator may result in artifacts and pattern collapse problems in practical applications.

Based on the above researches, hybrid models combining different functional networks have been proposed. One outstanding representative among these hybrid models is CVAE-GAN [12], which combines encoder, generator, classifier and discriminator together to build the model. Since the results from the generator of VAE [1] are usually blurry, naive combination of VAE and GAN does not take effect because they are easy tasks for the discriminator of GAN. In order to build an effective hybrid model, CVAE-GAN designs mean feature matching between various images produced by different functional network to alleviate gradient vanishing and pattern collapse. By means of abundant loss functions, CVAE-GAN is capable of generating samples with fine-grained category, however, with the price of a complex model, loss functions with lengthy items and more hyper-parameters. Actually, we find CVAE-GAN is more difficult to train compared to other simpler models in our experiments, and we failed to generate promising results on small datasets.

Recently, Introspective Variational Autoencoder (IntroVAE) [13] is proposed for synthesizing images. Its encoder and generator are trained in an introspective way, therefore, the encoder can somehow take the place of a discriminator. IntroVAE preserves the advantages of VAE models, such as stable training and synthesis diversity, and its inherent adversarial process makes it able to generate clearer images. Inspired by IntroVAE, we propose an introspective framework for conditional image synthesis. Our framework consists of three parts: encoder for inference and discrimination, generator for reconstruction and generation, and classifier for facilitating conditional synthesis. Moreover, the classifier helps to alleviate artifacts, meanwhile provides auxiliary feature measures for anti-blur between real images and reconstructed images.

In our structure of conditional introspective variational autoencoder, the encoder network undertakes more complex tasks than the others: it not only encodes the complicated original image distribution into the latent variable space, but also assumes the role of a discriminator to play introspectively adversarial game. With regard to the key role of encoder, we attempt to introduce some kind of new network for exploration. Capsule network (CapsNet) presented by Hinton [14] arouses our attention for its interesting characteristics. Capsule structures pack status of detected features in the form of vectors, which provide novel and more reasonable ways for mining features. Moreover, the dynamic routing between different capsule levels finds a new approach to integrate low level features into high level ones. In view of the above, we try to employ capsule structure in the encoder network. To our knowledge, capsule network has been adopted as a discriminator of GAN-type generative models in some literatures [15], [16], since this implementation is quite straightforward because CapsNet itself is originally a classification network. Whereas it is the first work that capsule is employed in the encoder of an image synthesis model. Certainly we adopt some specific approach to control conditional encoding, meanwhile we believe this processing will make full use of the capsule structure and exhibit potential results. Nevertheless, we also notice that the current implementation of capsule is far from perfection. As reported in some literatures [15]–[17], the original CapsNet only outperforms other convolutional neural networks (CNNs) on simple data sets such as MNIST, but fails to mine promising features on more complex ones. Although the capacity of present capsules are limited which may only make progress on the simplest hand-written digit synthesis, we believe this novel idea is of great potential and worthy of investigation. Our contribution can be concluded as followings:

(1) We design an end-to-end conditional generative model based on IntroVAE which contains three functional networks. Encoder and generator are adversarially trained in an introspective manner, and classifier is added for conditional synthesis. It is the first work that applies the introspective manner to conditional image synthesis as far as we know.
(2) Since classifier is added in this framework, we introduce feature matching for loss evaluations. Features captured by the classifier provide a good measure between the generated and the real images, therefore clearer images can be obtained.
(3) We investigate the capsule structure to construct the encoder, and design a class-wise capsule mask approach to control the label participation. In addition, we supplement capsule loss for encoder to further facilitate conditional encoding and avoid confusion between classes. Experimental results have shown the better clarity and some unique styles of the capsule encoding.

II. BACKGROUND AND RELATED WORK

A. BACKGROUND

Image synthesis can be divided into two categories: the unconditional and the conditional. The unconditional synthesis have firstly been researched and made progress. Deep Convolutional Generative Adversarial Network (DCGAN) [18] introduces the convolutional network into the generative model for unsupervised training. This structure uses the convolutional network to extract the feature and improve the learning effect of generator network. The dynamic routing used in [14] is regarded as a more robust algorithm for feature globalization compared to pooling used by CNNs, so some researches use CapsNet as the discriminator in place of CNN to synthesize more diverse and visually accurate images on the basis of DCGAN [15]. To stabilize the training process of GANs, Wasserstein GAN (WGAN) [19] utilizes the Earth Mover Distance as the objective for training the model.

Compared with the unconditional, conditional image synthesis can generate images according to some specific conditions. The provided conditions can be object category [11], attribute [9], description [8], image [4], etc. Conditions are usually given or learned as binary codes or embedding features [4], [8], [9], [11]. Recently, many generative models [10], [11], [15], [20]–[23] combine the conditions with the latent vector as the input of models. Based on VAE, CVAE [10] is developed to model complex structured output representations for structured output prediction. To conditionally generate images, CVAE modifies the conditional probability with condition labels for the inputs of both encoder and decoder. CGAN [11] is a conditional version of GAN by which some conditions are introduced into the generator and discriminator. Similar to CGAN, CDCGAN [20], [21] and CWGAN [22], [23] are proposed for conditional image synthesis based on GANs [18], [19]. PixelCNN provides a new approach based on image contexts [7], but its slow training speed and poor synthesis quality show that this type of model is far from maturity.

There are also some hybrid models by combining VAE and GAN. VAE/GAN [24] and Adversarial Autoencoders (AAE) [25] train the encoder, generator and discriminator to achieve image generation under unsupervised learning conditions, while the former utilizes the discriminator in data space and the latter utilizes the discriminator in latent space. Furthermore, ALI [26] and BiGAN [27] utilize the discriminators in both data space and latent space. Based on the architecture of VAE/GAN, CVAE-GAN [12] has added a new classifier and category labels to achieve fine-grained image generation, and simultaneously the mean feature matching is introduced into the loss function to avoid the model collapse problem to some extent. Moreover, IntroVAE [13] proposes a novel introspective variational autoencoder which can self-evaluate the quality of generated samples. IntroVAE contains an encoder network and a generative network, which form an adversarial relationship without extra discriminators.

B. RELATED WORK

The models which are most closely related to and have partly inspired our work are reviewed as following:

1) CVAE-GAN

The framework of CVAE-GAN consists of four parts: encoder, generator, discriminator and classifier [12]. The functions of its encoder and generator are the same as CVAE, while the discriminator and generator compete like mean feature matching based GAN. The mean feature matching between real samples and synthesized images requires an objective for the generator which is related to the discriminator. It is formulated as:

$$L_{GD} = \frac{1}{2}\left\| E_{x\sim P_r} f_D(x) - E_{z\sim P_z} f_D(G(z)) \right\|_2^2 . \tag{1}$$

where $f_D(x)$ denote features on an intermediate layer of the discriminator. Similarly, the classifier network and the reconstruction images are also involved in loss functions L_{GC} and L_G with the feature matching idea. Therefore, the final objective of CVAE-GAN can be assembled with:

$$L = L_D + L_C + \lambda_1 L_{KL} + \lambda_2 L_G + \lambda_3 L_{GD} + \lambda_4 L_{GC} \tag{2}$$

where L_D and L_C are loss functions of the discriminator and classifier, L_{KL} is related to encoder, L_G, L_{GD}, L_{GC} are related to the generator, and λs are four hyperparameters to balance each item.

2) INTROVAE

IntroVAE selects the encoder of VAEs as the discriminator of GANs and the generator of VAEs as the generator of GANs [13]. The objective of VAE is to maximize the variational lower bound of $p_\theta(x)$ as the following:

$$\begin{aligned} log p_\theta(x) \geq\ & E_{q_\phi(z|x)} log p_\theta(x|z) \\ & - D_{KL}\left(q_\phi(z|x)\,||p(z)\right). \end{aligned} \tag{3}$$

where this evidence lower bound (ELBO) objetive can be divided into two components: reconstruction error L_{AE} and the regularization term L_{REG}. L_{REG} is used as the adversarial training cost function, thus the encoder is trained to minimize L_{REG} to encourage the posterior $q_\phi(z|x)$ to match the prior $p(z)$, as well as to maximize L_{REG} to encourage the posterior $q_\phi\left(z|G(z')\right)$ to deviate from the prior $p(z)$. Conversely, the generator is trained to produce samples that have a small L_{REG}. Therefore the losses for the model are designed as:

$$L_E(x,z) = E(x) + [m - E(G(z))]^+ , \tag{4}$$

$$L_G(z) = E(G(z)) . \tag{5}$$

where $E(x) = D_{KL}\left(q_\phi(z|x)\,||p(z)\right)$, m is a positive margin, and $[\cdot]^+ = max(0,\cdot)$. To solve the latent mode collapse and training instability problems with the above adversarial manner, L_{AE} is combined with Eq. (4) and (5) respectively. The addition of reconstruction error builds a bridge between the encoder and the generator and usually makes the hybrid model more stable.

3) CAPSULE NETWORK (CAPSNET)

CapsNet is presented [14] where a capsule represents a group of neurons. The length of capsule's activity vector is used to represent the probability that the entity exists and an iterative routing-by-agreement mechanism is designed to improve the performance of CapsNet. CapsNet uses margin loss to evaluate the length of instantiation vector which represents the probability that an entity exists. This margin loss is defined as:

$$\begin{aligned} L_c = & T_c max\left(0, m^+ - \|v_c\|\right)^2 \\ & + \lambda\left(1 - T_c\right) max\left(0, \|v_c\| - m^-\right)^2 , \end{aligned} \tag{6}$$

where $T_c = 1$ marks the presence of a class c, $m^+ = 0.9$, $m^- = 0.1$ set the length thresholds, and $\lambda = 0.5$ down-weights the loss for absent classes.

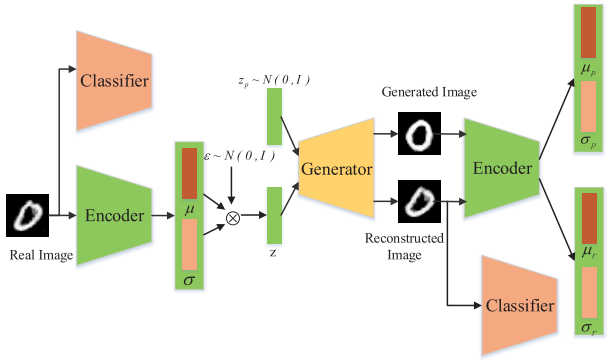


FIGURE 1. The architecture of our proposed conditional synthesis model. It consists of three components, the encoder network, the generative network and the classification network. Real images are encoded as latent variables by the encoder, and the generative network learns how to reconstruct images from the latent variables and generate image from random latent samples. Synthesized images are then fed into the encoder to obtain samples' posterior distribution, which will be matched to the prior distribution in the training. The real images and generated images are respectively fed into the classification network for measuring the classification probability and promoting the conditional generation effect.

III. METHODOLOGY

A. ARCHITECTURE

To conditionally generate clear and realistic images, the variational autoencoder learning framework with introspective training is proposed. As shown in Figure 1, the present architecture consists of three components: an encoder, a generator and a classifier.

The encoder is used to map an input image x into an approximate posterior to match the prior distribution, so does the encoder network in CVAE which employs traditional CNNs. Furthermore, similar to the one in IntroVAE, the encoder here also provides an adversarial mode and then plays a role in discriminating the generated samples from the training data. This contributes to generate sharp images.

In our pipeline, we have implemented two different structures for the encoder. The first one simply employs a traditional CNN as the encoder network to evaluate the effect of our framework. Considering the conditional encoding potential of capsule structure, a Capsule Network (CapsNet) is employed for an optional implementation. As far as we know, different from the current work which uses CapsNet as the discriminator or classifier, this study is the first work to explore the encoding capacity of CapsNet. The dynamic routing between capsules is a more robust feature integration scheme than the traditional max-pooling of CNNs, which makes high-level capsules have stronger feature expression ability. Moreover, we are able to achieve conditional encoding by masking high-level capsules of other categories with the employment of capsules. This is different from other works in which class labels and input images are encoded after concatenating them. In order to distinguish our models realized by two different encoders, we name the former as CCVAE and the latter as CCapsVAE in the following expression.

B. CONDITIONAL INTROSPECTIVE ENCODER

Multiple loss functions are designed for the encoder of CCVAE to achieve conditional introspective training. First of all, the KL-divergence [28], a classical mathematical tool which can be used to approximate very complex data distributions, is employed to regularize the encoder by encouraging the posterior probability $q_\phi(z|x)$, where z is the latent vector corresponding to the input image x, to match the prior probability $p(z)$. Following the original VAEs [1], we use the centered isotropic multivariate Gaussian $N(0, I)$ to describe the prior probability. Let the posterior probability denote by

$$q_\phi(z|x) = N(z; \mu, \sigma^2), \tag{7}$$

where μ and σ are the mean and variance, respectively, computed from the output variables of the encoder network. The input z of the generator G is generally sampled from $N(\mu, \sigma^2)$. Note that the sampling operation is discrete and non-differentiable and thus it is hard to perform the gradient-based back propagation. To address this point, the method of reparameterization trick [1] is used here, according to which a random variable ϵ is first sampled from a Gaussian distribution and then translated and zoomed to z, as the following:

$$z = \mu + \epsilon \odot \sigma, \tag{8}$$

where $\epsilon \sim N(0, I)$, and $\odot$ means the element-wise multiplication. Then given N data samples, the KL-divergence L_{KL}, as the first sub-objective of the encoder, can be computed as below:

$$L_{KL}(z; \mu, \sigma) = \frac{1}{2} \sum_{i=1}^{N} \left(1 + log\left({\sigma_i}^2\right) - {\mu_i}^2 - {\sigma_i}^2\right). \tag{9}$$

For conditional image generation, the feature of the a generated image which belongs to a specific category should be matched with the average feature of real data in this category [29]. According to this idea, the feature matching objective, denoted by L_{FM} here, is used to make the generated images as distinct and similar as possible to the real data. Moreover, it also prevents the model from gradient vanishing and makes the training process more stable. As the second sub-objective, L_{FM} is composed of two components. The first component, denoted by $L_{mse}(x_r, r)$, is expressed by the pixel-wise mean squared error (MSE) function, which measures the difference between the real image x and the reconstructed image x_r, as following:

$$L_{mse}(x_r, x) = \frac{1}{2} \sum_{i=1}^{N} \left\| x_{r,i} - x_i \right\|_F^2, \tag{10}$$

where $x_{r,i}$ denotes the reconstruction of the i-th data x_i. A smaller $L_{mse}(x_r, r)$ suggests that x_r and x are closer to each other. The second component of L_{FM} is the mean feature matching, denoted by $L_{mfm}(x_r, x)$, similar to the expression in literature [12], which requires that the feature center of the reconstructed image x_r to match the feature center of the real

image x, as following:

$$L_{mfm}(x_r, x) = \frac{1}{2}\sum_{i=1}^{N}\left\|f_C\left(x_{r,i}\right) - f_C\left(x_i\right)\right\|_2^2, \tag{11}$$

where the function $f_C(\cdot)$ returns the features on an intermediate layer of the classification network. For simplicity, we select the input of the last fully connected (FC) layer of classification network as the features. Then, based on Formulas (10) and (11), the final feature matching objective L_{FM} is defined as:

$$L_{FM}(x_r, x) = L_{mse}(x_r, x) + L_{mfm}(x_r, x). \tag{12}$$

Moreover, in the training process, two types of fake samples are used to learn more expressive latent features and generate more realistic images [30]. They are the reconstruction samples x_r from the posterior $q_\phi(z|x)$ and the generated samples x_p from the prior $p(z)$, respectively. For training the model in an adversarial manner, we use the KL-divergence L_{KL} defined in Formula (9) as the cost function of the adversarial training. The encoder network is trained to minimize $L_{KL}(z)$ and maximize $L_{KL}(z_s)$, where z and z_s denote the latent vectors corresponding to x and x_s $(s = r, p)$, respectively. Therefore, the total loss function for the encoder network of CCVAE is defined as:

$$\begin{aligned} L_E = {} & L_{KL}(z) + \alpha \sum_{s=r,p}[m - L_{KL}(z_s)]^+ \\ & + \beta L_{FM}(x, x_r), \end{aligned} \tag{13}$$

where $[\cdot]^+ = max(0, \cdot)$, α and β are the weighting parameters used to balance the importance of each item, and m is a positive margin. When $L_{KL}(z_s) \leq m$, Formulas (13) and (15) form a min-max game between the encoder and the generator.

C. CONDITIONAL CAPSULE ENCODER

If the encoder of our model is implemented with the capsule network, a novel approach of label participation in conditional encoding can be employed. Figure 2 demonstrates the mechanism of the conditional capsule encoder for CCapsVAE. High-level capsules are masked by the class label to obtain conditional capsule features. Except the specified capsule, all vectors are masked to zero.

For the encoder of CCapsVAE, there are four sub-objectives jointly trained. Based on the above subsection, the capsule loss L_c in Formula (6) is added to facilitate conditional synthesis. Therefore, the total loss function for the encoder of CCapsVAE is defined as:

$$\begin{aligned} L_E = {} & L_{KL}(z) + \alpha \sum_{s=r,p}[m - L_{KL}(z_s)]^+ \\ & + \beta L_{FM}(x, x_r) + L_c, \end{aligned} \tag{14}$$

where notations are the same as Formula (13). The first two components guarantee the diversity of synthesis and self-evaluation, L_{FM} enables the reality of generation, and L_c improves the conditional capsule encoding.

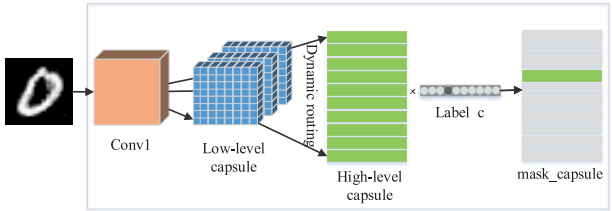


FIGURE 2. Mechanisms of the conditional capsule encoder.

D. GENERATOR AND CLASSIFIER

The implementations of the generator and classifier of CCVAE and CCapsVAE are coincident. For the generator, to make the posterior distribution of the generated / reconstructed images match the prior distribution, the training objective is defined as the following:

$$L_G = \alpha \sum_{s=r,p} L_{KL}(z_s) + \beta L_{FM}(x, x_r). \tag{15}$$

The symbols in Formula (15) have the same meanings to those in the above-mentioned formulas. The feature matching based reconstruction loss is also added into this objective to offer introspective competition and promote more realistic generation of the generator.

For the classifier, it is implemented by a traditional CNN-based classification network which is optimized by the cross-entropy loss function. The features before the last fully connected layer of the classifier are used for feature matching which facilitates for conditional image generation as mentioned above. The details of generator and classifier will be expounded in Section IV-A.

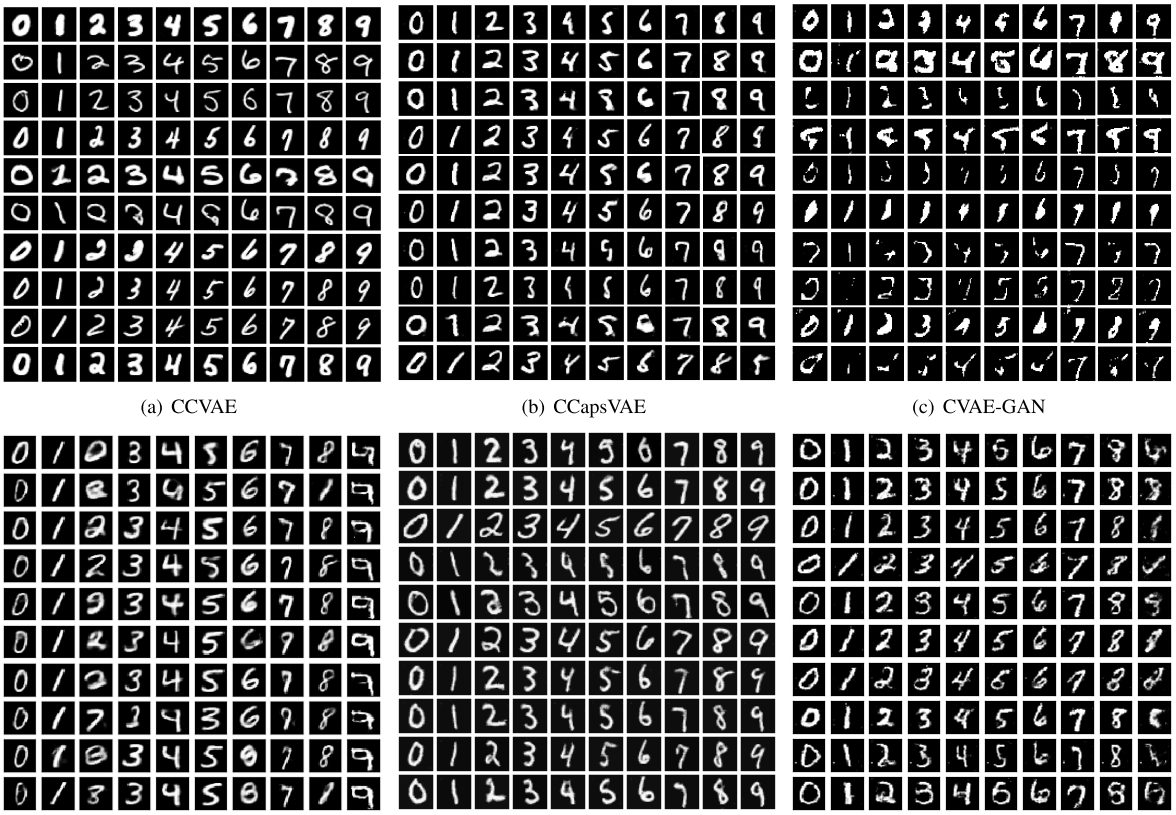
IV. EXPERIMENTS

A. IMPLEMENTATIONS

On account of our computational power and the estimated capacity of capsule network, the proposed model is evaluated on two small datasets: MNIST and Fashion-MNIST. We compare our models with several typical conditional generative models including state-of-the-art ones. MNIST includes the handwritten digital data with 10 categories. And Fashion-MNIST covers a total of 70,000 images from 10 different categories, with size, format, and training / test set partition as same as those of the original MNIST dataset. Each image of these two datasets is composed of 28×28 pixels, and each pixel is represented by a grayscale value. So the size of every synthesized image is 28×28 as well.

In the experiments, the encoder of CCVAE consists of four convolutional layers, followed by two fully connected layers, which have 128, 256, 256 and 512 channels with the filter size of 5×5, respectively. All of these convolutional layers use the stride of 2 and the Leaky ReLU activation.

And the encoder of CCapsVAE uses a Capsule Network with two convolutional layers and one fully connected layer. The first convolutional layer includes 256 convolution kernels with the size of 9×9, and every kernel has the stride of 1 and the ReLU activation. The second convolutional layer is



(a) CCVAE (b) CCapsVAE (c) CVAE-GAN

(d) CVAE (e) CWGAN (f) CDCGAN

FIGURE 3. Images conditionally generated by the six models, trained over MNIST. (a) CCVAE, (b) CCapsVAE, (c) CVAE-GAN, (d) CVAE, (e) CWGAN, and (f) CDCGAN. CCapsVAE is trained for only 11 epochs, CDCGAN for 30, CCVAE and CVAE-GAN for 50, and CVAE and CWGAN for 100 epochs.

regarded as the Primary Capsule Layer [14] of CapsNet. Here $32 \times 6 \times 6$ capsule outputs are produced and each capsule is an 8-dimensional vector. These capsules are routed to the final fully connected layer by the iterative routing-by-agreement mechanism. The final layer generates ten 16-dimensional capsules. Then, these high-level capsules are masked by the class label for conditional encoding.

The generator consists of a fully connected layer, followed by five deconvolutional layers. Among these deconvolutional layers, all of the first four use the same stride of 2 and the Leaky ReLU activation, whereas they produce 512, 256, 256 and 128 channels with the filter size of 5×5, respectively. The last deconvolutional layer employs one 5×5 convolution kernel with the stride of 1 and the tanh activation.

For the classifier, four convolutional layers are employed, followed by two fully connected layers. The last layer outputs the classification probability by a softmax function.

To balance the training of the encoder and generator models, the configuration of the hyper-parameters is: $m = 10, \alpha = 0.5, \beta = 0.25$.

B. RESULTS AND ANALYSIS

In this subsection, the experimental results on MNIST and Fashion-MNIST are exhibited and discussed from both visual and quantitative aspects.

1) VISUAL RESULTS AND ANALYSIS

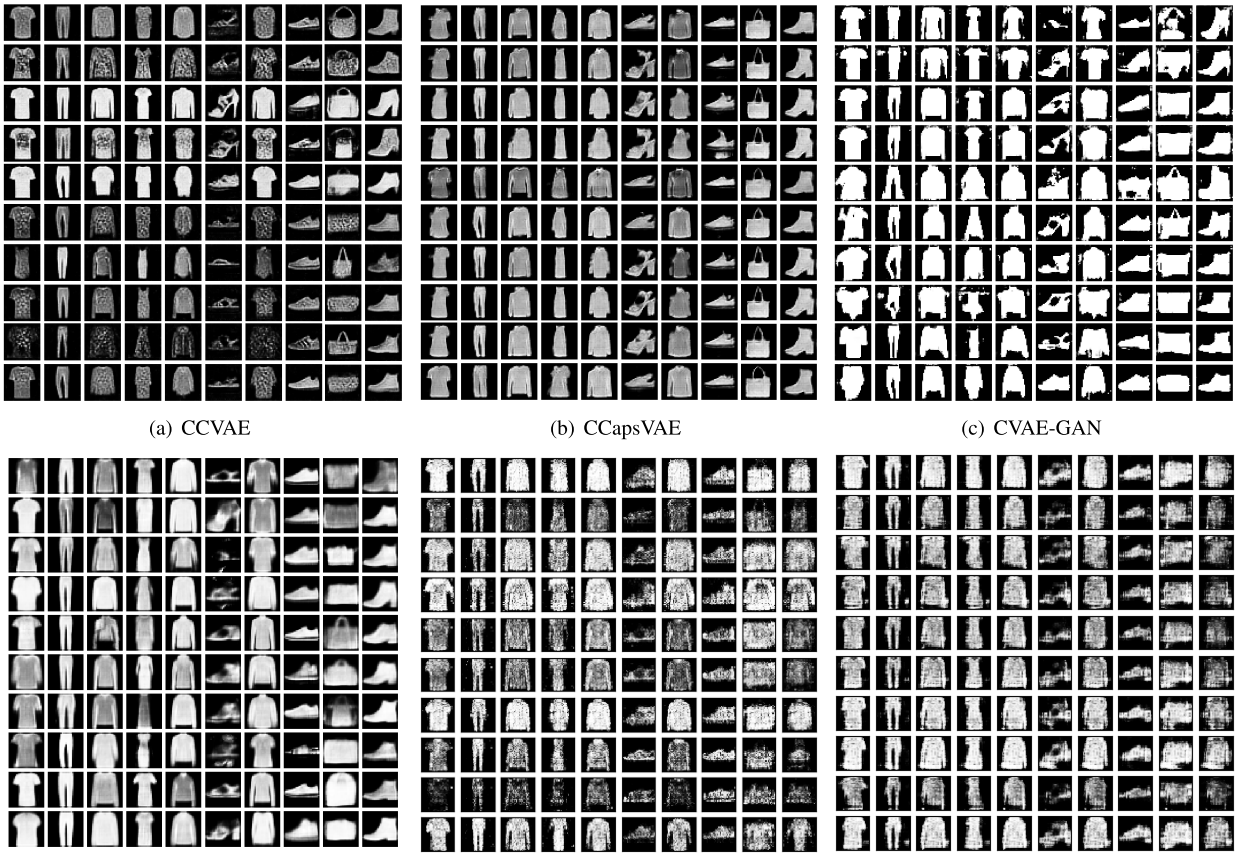
From the results shown in Figure 3, we can draw the following observations:

(1) **Realness and artifacts.** CCVAE can produce more realistic images with few artifacts than the existing methods. CCapsVAE and CWGAN also perform well with delicate strokes and fewer artifacts. Images from CDCGAN are unnatural to some extent, and the most severe artifacts of CVAE-GAN are attributed to the unsuccessful training.

(2) **Clarity and blur.** Actually, images with the finest strokes and sharpest contours are observed from CCapsVAE, which acts more like GANs. CCVAE generates images with less blur than CVAE, confirming that the feature matching has compensated for the deficiency of MSE.

(3) **Synthesis diversity.** Diversity of CCVAE is the best among the involved competitors, where different writing styles can be observed in the conditional generation. Synthesized digits of ambiguous classes are also the fewest, which proves the superiority of the proposed conditional introspective framework.

(4) **Training difficulty.** Although the CapsNet is shallow with only three layers, the CCapsVAE can learn the features of images faster and generate good-quality



(a) CCVAE (b) CCapsVAE (c) CVAE-GAN

(d) CVAE (e) CGAN (f) CDCGAN

FIGURE 4. Conditional images generated by (a) CCVAE, (b) CCapsVAE, (c) CVAE-GAN, (d) CVAE, (e) CGAN and (f) CDCGAN, trained over Fashion-MNIST.

results with fewer training iterations compared to other models. CVAE-GAN is the most difficult to train among all the models in our experiments. As shown in Figure 3(c), we failed to generate promising results by configurating hyper-parameters with our best efforts.

Results on the Fashion-MNIST dataset are shown in Figure 4. For all models, we conduct experiments without tuning any hyper-parameters. As illustrated in Figure 4(a) and 4(b), the images generated by CCVAE and CCapsVAE are clear and realistic enough. On this data set, CCVAE shows a comprehensive synthesis performance, with both good diversity and completeness of patterns. CCapsVAE is able to produce clear patterns, but diversity is somehow inferior to CCVAE. Textures and patterns on clothing are also ignored by the capsule encoder, which tends to generate clear images with homogenous gray levels. All GAN-type models have poor conditional generation performances on this data set. Figure 4(e) and Figure 4(f) show the results of CGAN and CDCGAN after 100 epochs of training. In their columns labeled "Ankle boot", many other objects appear including "T-shirt", "Pullover" and "Bag", but patterns like "Ankle boot" does not. Furthermore, the image artifacts generated by CGAN and CDCGAN are very obvious. CVAE shows a blurry but not bad result as we expect, and CVAE-GAN is again difficult to train. Based on the above visual results, our proposed framework has obvious advantages compared to other baseline methods, while the capsule encoder seems to be more suitable for structural feature capture.

2) QUANTITATIVE RESULTS AND ANALYSIS

In addition to the visual observation of experimental results, the popular inception score [31] is employed as a quantitative evaluation measure. Since the inception score evaluates the image quality in terms of diversity and clarity, it is actually inadequate to reflect the effects of conditional synthesis. However, due to the lack of commonly accepted metrics for evaluating whether the generated image is well conditioned, we just supplement the clarity calculation as another synthesis quality metric for reference. The results on MNIST and Fashion-MNIST are shown in Table 1, according to which we have the following observations:

(1) On both data sets, the indicators of CCVAE and CCapsVAE take the leading places, which validate the effectiveness of our proposed models.

TABLE 1. Quantitative results on MNIST and Fashion-MNIST.

Tested Models \ Dataset	MNIST		Fashion-MNIST	
	Inception	Clarity	Inception	Clarity
Real image	9.8793 ± 0.0614	0.9919 ± 0.0037	9.0617 ± 0.0430	0.9100 ± 0.0041
CVAE	2.0594 ± 0.0426	0.7232 ± 0.0048	3.5721 ± 0.0483	0.5507 ± 0.0044
CGAN	1.1373 ± 0.0012	0.8382 ± 0.0008	3.1818 ± 0.0640	$\mathbf{0.6397 \pm 0.0062}$
CDCGAN	2.0851 ± 0.0800	0.8300 ± 0.0056	2.7338 ± 0.0464	0.6070 ± 0.0061
CWGAN	2.5042 ± 0.0820	0.7905 ± 0.0073	—	—
CCVAE	$\mathbf{2.6463 \pm 0.1007}$	0.7985 ± 0.0093	3.4170 ± 0.1455	0.5339 ± 0.0073
CCapsVAE	2.2970 ± 0.0512	$\mathbf{0.8468 \pm 0.0048}$	$\mathbf{4.1865 \pm 0.0627}$	0.6137 ± 0.0042

(2) Overall, higher inception scores seem to indicate better synthesis diversity. However, this diversity may only reflect an overall synthesis effect, not for each condition. For example, CVAE obtains low clarity values but fine inception scores on both data sets, which mean the diversity is good under the measure of metrics. But if we refer to the visual results, many samples of ambiguous classes will be observed. Moreover, the clarity metric is partial to GAN-type models in spite of their artifacts. Although our models show pleasant results on the quantitative measures, appropriate metrics for conditional synthesis still remain to be studied.

V. CONCLUSION

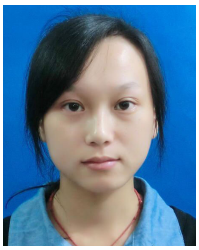
In this paper, we propose a conditional variational autoencoder for image synthesis. A framework consisting of three parts is trained in an introspective manner to produce realistic, clear and diverse images. Moreover, the capsule structure is investigated as the encoder to achieve conditional encoding, and experimental results have shown us a surprisingly better synthesis clarity than we imagine. Our future work will attempt to realize more complex and realistic image synthesis of high-resolution. And the encoding of real image distribution and mining of image essential features for image generation will be further studied.

REFERENCES

[1] D. P Kingma and M. Welling, "Auto-encoding variational bayes," 2013, *arXiv:1312.6114*. [Online]. Available: http://arxiv.org/abs/1312.6114
[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[3] L. Wang, W. Chen, W. Yang, F. Bi, and F. R. Yu, "A State-of-the-Art review on image synthesis with generative adversarial networks," *IEEE Access*, vol. 8, pp. 63514–63537, 2020.
[4] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.
[5] Y. Tian, X. Peng, L. Zhao, S. Zhang, and D. N. Metaxas, "CR-GAN: Learning complete representations for multi-view generation," 2018, *arXiv:1806.11191*. [Online]. Available: http://arxiv.org/abs/1806.11191
[6] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning Text-To-Image generation by redescription," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1505–1514.
[7] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, and A. Graves, "Conditional image generation with pixeLCNN decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 4790–4798, 2016.
[8] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1316–1324.
[9] H. Huang, L. Song, R. He, Z. Sun, and T. Tan, "Variational capsules for image analysis and synthesis," 2018, *arXiv:1807.04099*. [Online]. Available: http://arxiv.org/abs/1807.04099
[10] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3483–3491.
[11] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*. [Online]. Available: http://arxiv.org/abs/1411.1784
[12] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "CVAE-GAN: Fine-grained image generation through asymmetric training," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2745–2754.
[13] H. Huang, R. He, Z. Sun, and T. Tan, "IntroVAE: Introspective variational autoencoders for photographic image synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 52–63.
[14] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3856–3866.
[15] Y. Upadhyay and P. Schrater, "Generative adversarial network architectures for image synthesis using capsule networks," 2018, *arXiv:1806.03796*. [Online]. Available: http://arxiv.org/abs/1806.03796
[16] A. Jaiswal, W. AbdAlmageed, Y. Wu, and P. Natarajan, "CapsuleGAN: Generative adversarial capsule network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 526–535.
[17] V. M. do Rosario, E. Borin, and M. Breternitz, "The multi-lane capsule network," *IEEE Signal Process. Lett.*, vol. 26, no. 7, pp. 1006–1010, Jul. 2019.
[18] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*. [Online]. Available: http://arxiv.org/abs/1511.06434
[19] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*. [Online]. Available: http://arxiv.org/abs/1701.07875
[20] P. L. Suárez, A. D. Sappa, and B. X. Vintimilla, "Colorizing infrared images through a triplet conditional DCGAN architecture," in *Proc. Int. Conf. Image Anal. Process. (ICIAP)*, Catania, Italy. Cham, Switzerland: Springer, 2017, pp. 287–297.
[21] K. Sricharan, R. Bala, M. Shreve, H. Ding, K. Saketh, and J. Sun, "Semi-supervised conditional GANs," 2017, *arXiv:1708.05789*. [Online]. Available: http://arxiv.org/abs/1708.05789
[22] Y. Zhu, M. Aoun, M. Krijn, J. Vanschoren, and H. T. Campus, "Data augmentation using conditional generative adversarial networks for leaf counting in arabidopsis plants," in *Proc. BMVC*, 2018, p. 324.
[23] Y. Cao, B. Liu, M. Long, and J. Wang, "HashGAN: Deep learning to hash with pair conditional wasserstein GAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1287–1296.
[24] A. Boesen Lindbo Larsen, S. Kaae Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," 2015, *arXiv:1512.09300*. [Online]. Available: http://arxiv.org/abs/1512.09300
[25] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," 2015, *arXiv:1511.05644*. [Online]. Available: http://arxiv.org/abs/1511.05644
[26] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," 2016, *arXiv:1605.09782*. [Online]. Available: http://arxiv.org/abs/1605.09782
[27] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," 2016, *arXiv:1606.00704*. [Online]. Available: http://arxiv.org/abs/1606.00704
[28] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.

[29] M. Pieters and M. Wiering, "Comparing generative adversarial network techniques for image creation and modification," 2018, *arXiv:1803.09093*. [Online]. Available: http://arxiv.org/abs/1803.09093
[30] S. Zhao, J. Song, and S. Ermon, "InfoVAE: Information maximizing variational autoencoders," 2017, *arXiv:1706.02262*. [Online]. Available: http://arxiv.org/abs/1706.02262
[31] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.

KUN ZHENG received the Ph.D. degree from the China University of Geosciences, Wuhan, China. He is currently an Associate Professor with the School of Geography and Information Engineering, China University of Geosciences. His current research interests involve spatio-temporal data visual analytics, storage and management of spatial big data.



YAFAN CHENG received the B.S. degree in network engineering from the China University of Geoscience, Wuhan, China, in 2018, where she is currently pursuing the master's degree in computer science and technology. Her main research interests include image classification and image generation.

XIAOJUN KANG (Member, IEEE) received the B.S. degree in computer science and technology from the China University of Geosciences, China, in 2000, and the Ph.D. degree in resource and environment information engineering from Huazhong Agricultural University, China, in 2011. He is currently an Associate Professor with the School of Computer Science, China University of Geosciences. His research interests include artificial intelligence, natural language processing, and bioinformatics.

HONG YAO (Member, IEEE) received the B.S. degree in computer and applications from the Wuhan Technical University of Surveying and Mapping, China, in 1998, and the Ph.D. degree in computer science and technology from the Huazhong University of Science and Technology, China, in 2010. He is currently a Professor with the School of Computer Science, China University of Geosciences, China. His research interests include knowledge graph, wireless and mobile networks, and mobile cloud computing.

TIAN TIAN (Member, IEEE) received the B.S. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2009 and 2015, respectively. She joined the School of Computer Sciences, China University of Geosciences, Wuhan, China, in 2015, as a Postdoctoral Lecturer. She is currently an Associate Professor with the School of Computer Sciences. Her major research interests include remote sensing image processing, and computer vision and applications.