CDGAN: Cyclic Discriminative Generative Adversarial Networks for Image-to-Image Transformation

Kancharagunta Kishan Babu and Shiv Ram Dubey

Abstract—Generative Adversarial Networks (GANs) have facilitated a new direction to tackle the image-to-image transformation problem. Different GANs use generator and discriminator networks with different losses in the objective function. Still there is a gap to fill in terms of both the quality of the generated images and close to the ground truth images. In this work, we introduce a new Image-to-Image Transformation network named Cyclic Discriminative Generative Adversarial Networks (CDGAN) that fills the above mentioned gaps. The proposed CDGAN generates high quality and more realistic images by incorporating the additional discriminator networks for cycled images in addition to the original architecture of the CycleGAN. The proposed CDGAN is tested over three image-to-image transformation datasets. The quantitative and qualitative results are analyzed and compared with the state-of-the-art methods. The proposed CDGAN method outperforms the state-of-the-art methods when compared over the three baseline Image-to-Image transformation datasets. The code is available at https://github.com/KishanKancharagunta/CDGAN.

Index Terms—Deep Networks; Computer Vision; Generative Adversarial Nets; Image-to-Image Transformation; Cyclic-Discriminative Adversarial loss;

I. INTRODUCTION

There are many real world applications where images of one particular domain need to be translated into different target domain. For example, sketch-photo synthesis [1], [2], [3] is required in order to generate the photo images from the sketch images that helps to solve many real time law and enforcement cases, where it is difficult to match sketch images with the gallery photo images due to domain disparities. Similar to sketch-photo synthesis, many image processing and computer vision problems need to perform the image-to-image transformation task, such as Image Colorization, where grayscale image is translated into the colored image [4], [5]. Image in-painting, where lost or deteriorated parts of the image are reconstructed [6], [7]. Image, video and depth map super-resolution, where resolution of the images is enhanced [8], [9]. Artistic style transfer, where the semantic content of the source image is preserved while the style of the target image is transferred to the source image [10], [11], and Image denoising, where the original image is reconstructed from the noisy measurement [12]. Some other applications like rain or haze removal from the images [13], [14], [15], deblurring [16].

Radial Distortion Rectification [17], visualization [18], [19], cross-modal representation [20], generating realistic videos [21], predicting skeletal activity for early activity recognition [22] are also needed to perform image-to-image transformation. However, traditionally the image-to-image transformation methods are proposed for a particular specified task with the specialized method, which is suited for that task only.

A. CNN Based Image-to-Image Transformation Methods

Deep Convolutional Neural Networks (CNN) are used as an end-to-end frameworks for image-to-image transformation problems. The CNN consists of a series of convolutional and deconvolutional layers. It tries to minimize the single objective (loss) function in the training phase while learns the network weights that guide the image-to-image transformation. In the testing phase, the given input image of one visual representation is transformed into another visual representation with the learned weights. For the first time, Long et al. [23] have shown that the convolutional networks can be trained in end-to-end fashion for pixelwise prediction in semantic segmentation problem. Larson et al. [24] have...
developed a fully automatic colorization method for translating the grayscale images to the color images using a deep convolutional architecture. Zhang et al. have proposed a novel method for end-to-end photo-to-sketch synthesis using the fully convolutional network. Gatys et al. have introduced a neural algorithm for image style transfer, that constrains the texture synthesis with learned feature representations from the state-of-the-art CNNs. Feng et al. have investigated a Dual Swap Disentangling (DSD) weakly semi-supervised method for learning interpretable disentangled representations with the help of dual autoencoder structure. These methods treat image transformation problem separately and designed CNNs suited for that particular problem only. It opens an active research scope to develop a common framework that can work for the different image-to-image transformation problems.

B. GAN Based Image-to-Image Transformation Methods

In 2014, Ian Goodfellow et al. have introduced Generative Adversarial Networks (GAN) as a general purpose solution to the image generation problem. Rather than generating images of the given random noise distribution as mentioned in, GANs are also used for different computer vision applications like image super resolution, real-time style transfers, sketch-to-photo synthesis, text-to-image synthesis, driver drowsiness detection or recognition, and domain adaptation. Mizra et al. have introduced conditional GANs (cGAN) by placing a condition on both of the generator and the discriminator networks of the basic GAN using class labels as an extra information. Conditional GANs have boosted up the image generation problems. Since then, an active research is being conducted to develop the new GAN models that can work for different image generation problems. Still, there is a need to develop the common methods that can work for different image generation problems, such as sketch-to-face, NIR-to-RGB, etc.

Isola et al. have introduced Pix2Pix, as a general purpose method consisting a common framework for the image-to-image transformation problems using Conditional GANs (cGANs). Pix2Pix works only for the paired image datasets. It consists the generative adversarial loss and L2 loss as an objective function. Wang et al. have proposed PAN, a general framework for image-to-image transformation tasks by introducing the perceptual adversarial loss and combined it with the generative adversarial loss. Zhu et al. have investigated CycleGAN, an image-to-image transformation framework that works over unpaired datasets also. The unpaired datasets make it difficult to train the generator network due to the discrepancies in the two domains. This leads to a mode collapse problem where the majority of the generating images share the common properties results in similar images as outputs for different input images. To overcome this problem Cycle-consistency loss is introduced along with the adversarial loss. Yi et al. have developed DualGAN, a dual learning framework for image-to-image transformation of the unsupervised data. DualGAN consists of the reconstruction loss (similar to the Cycle-consistency loss in) and the adversarial loss as an objective function. Wang et al. have introduced PS2MAN, a high quality photo-to-sketch synthesis framework consisting of multiple adversarial networks at different resolutions of the image. PS2MAN consists Synthesized loss in addition to the Cycle-consistency loss and adversarial loss as an objective function. Recently, Kanchargunta et al. have proposed CSGAN, an image-to-image transformation method using GANs. CSGAN considers the Cyclic-Synthesized loss along with the other losses mentioned in.

Most of the above mentioned methods consist of two generator networks G_{AB} and G_{BA} which read the input data from the Real_Images (R_A and R_B) from the domains A and B, respectively. These generators G_{AB} and G_{BA} generate the Synthesized_Images (S_{yn,B} and S_{yn,A}) in domain B and A, respectively. The same generator networks G_{AB} and G_{BA} are also used to generate the Cycled_Images (C_{yc,B} and C_{yc,A}) in the domains B and A from the Synthesized_Images S_{yn,A} and S_{yn,B}, respectively. In addition to the generator networks G_{AB} and G_{BA}, the above mentioned methods also consist of the two discriminator networks D_A and D_B. These discriminator networks D_A and D_B are used to distinguish between the Real_Images (R_A and R_B) and the Synthesized_Images (S_{yn,A} and S_{yn,B}) in the its respective domains. The losses, namely Adversarial loss, Cyclic-Consistency loss, Synthesized loss and Cyclic-Synthesized loss are used in the objective function.

In this paper, we introduce a new architecture called Cyclic-Discriminative Generative Adversarial Network (CDGAN) for the image-to-image transformation problem. It improves the architectural design by introducing a Cyclic-Discriminative adversarial loss computed among the Real_Images and the Cycled_Images. The CDGAN method also consists of two generators G_{AB} and G_{BA} and two discriminators D_A and D_B similar to other methods. The generator networks G_{AB} and G_{BA} are used to generate the Synthesized_Images (S_{yn,B} and S_{yn,A}) from the Real_Images (R_A and R_B) and the Cycled_Images (C_{yc,B} and C_{yc,A}) from the Synthesized_Images (S_{yn,A} and S_{yn,B}) in two different domains A and B, respectively. The two discriminator networks D_A and D_B are used to distinguish between the Real_Images (R_A and R_B) and the Synthesized_Images (S_{yn,A} and S_{yn,B}) and also between the Real_Images (R_A and R_B) and the Cycled_Images (C_{yc,A} and C_{yc,B}).

Following is the main contributions of this work:

- We propose a new method called Cyclic Discriminative Generative Adversarial Network (CDGAN), that uses Cyclic-Discriminative (CD) adversarial loss computed over the Real_Images and the Cycled_Images. This loss helps to increase the quality of the generated images and also reduces the artifacts in the generated images.
- We evaluate the proposed CDGAN method over three benchmark image-to-image transformation datasets with four different benchmark image quality assessment measures.
- We conduct the ablation study by extending the concept of proposed Cyclic-Discriminative adversarial loss between the Real_Images and the Cycled_Images with the
state-of-the-art methods like CycleGAN [27], DualGAN [26], PS2GAN [42] and CSGAN.

The remaining paper is arranged as follows: Section II presents the proposed method and the losses used in the objective function; Experimental setup with the datasets and evaluation metrics used in the experiment are shown in section III. Result analysis and ablation study are conducted in section IV followed by the Conclusions in section V.

II. PROPOSED CDGAN METHOD

Consider a paired image dataset $X$ between two different domains $A$ and $B$ represented as $X \in \{(A_i), (B_i)\}_{i=1}^n$ where $n$ is the number of pairs. The goal of the proposed CDGAN method is to train two generator networks $G_{AB} : A \rightarrow B$ and $G_{BA} : B \rightarrow A$ and two discriminator networks $D_A$ and $D_B$. The generator $G_{AB}$ is used to translate the given input image from domain $A$ into the output image of domain $B$ and the generator $G_{BA}$ is used to transform an input sample from domain $B$ into the output sample in domain $A$. The discriminator $D_A$ is used to differentiate between the real and the generated image in domain $A$ and in the similar fashion the discriminator $D_B$ is used to differentiate between the real and the generated image in domain $B$. The Real_Images ($R_A$ and $R_B$) from the domains $A$ and $B$ are given to the generators $G_{AB}$ and $G_{BA}$ to generate the Synthesized_Images ($Syn_B$ and $Syn_A$) in domains $B$ and $A$, respectively as,

$$Syn_B = G_{AB}(R_A)$$

(1)

$$Syn_A = G_{BA}(R_B).$$

(2)

The Synthesized_Images ($Syn_B$ and $Syn_A$) are given to the generators $G_{BA}$ and $G_{AB}$ to generate the Cycled_Images ($Cyc_A$ and $Cyc_B$), respectively as,

$$Cyc_A = G_{BA}(Syn_B) = G_{BA}((G_{AB}(R_A))$$

(3)

$$Cyc_B = G_{AB}(Syn_A) = G_{AB}(G_{BA}(R_B)).$$

(4)

where, $Cyc_A$ is the Cycled_Image in domain $A$ and $Cyc_B$ is the Cycled_Image in domain $B$. As shown in the Fig. 2, the overall work of the proposed CDGAN method is to read two Real_Images ($R_A$ and $R_B$) as input, one from the domain $A$ and another from the domain $B$. These images $R_A$ and $R_B$ are first translated into the Synthesized_Images ($Syn_B$ and $Syn_A$) of other domains $B$ and $A$ by giving them to the generators $G_{AB}$ and $G_{BA}$, respectively. Later the translated Synthesized_Images ($Syn_B$ and $Syn_A$) from the domains $B$ and $A$ are again given to the generators $G_{BA}$ and $G_{AB}$ respectively, to get the translated Cycled_Images ($Cyc_A$ and $Cyc_B$) in domains $A$ and $B$, respectively.

The proposed CDGAN method is able to translate the input image $R_A$ from domain $A$ into the image $Syn_B$ in another domain $B$, such that the $Syn_B$ has to be look like same as the $R_B$. In the similar fashion the input image $Real_B$ is translated into the image $Syn_A$, such that it also looks like same as the $R_A$. The difference between the input real images and the translated synthesized images should be minimized in order to get the more realistic generated images. Thus, the suitable loss functions are needed to be used.

A. Objective Functions

The proposed CDGAN method as shown in the Fig. 2 consists of five loss functions, namely Adversarial loss, Synthesized loss, Cycle-consistency loss, Cyclic-Synthesized loss and Cyclic-Discriminative (CD) Adversarial loss.

1) Adversarial Loss: The least-squares loss introduced in LSGAN [44] is used as an objective in Adversarial loss instead of the negative log likelihood objective in the vanilla GAN [33] for more stabilized training. The Adversarial loss is calculated between the fake images generated by the generator network against the decision by the discriminator network either it is able to distinguish it as real or fake. The generator network tries to generate the fake image which looks like

Fig. 2: Image-to-image transformation framework based on proposed Cyclic Discriminative Generative Adversarial Networks (CDGAN) method. $G_{AB}$ and $G_{BA}$ are the generators, $D_A$ and $D_B$ are the discriminators, $Real_A$ and $Real_B$ are the real images, $Syn_A$ and $Syn_B$ are Synthesized images, $Cyc_A$ and $Cyc_B$ are the Cycled images in domain $A$ and domain $B$, respectively.
same as the real image. The real and the generated images are distinguished using the discriminator network. In the proposed CDGAN, \( \text{Synthesized Image} \) \((\text{Syn}_B)\) in domain \( B \) is generated from the generator \( G_{AB} : A \rightarrow B \) by using the Real Image \((R_A)\) of domain \( A \). The Real Image \((R_B)\) and the \( \text{Synthesized Image} \) \((\text{Syn}_B)\) in domain \( B \) are distinguished by the discriminator \( D_B \). It is written as,

\[
\mathcal{L}_{\text{LSGAN}_B}(G_{AB}, D_B, A, B) = E_{B \sim P_{\text{data}}(B)}[(D_B(R_B) - 1)^2] + E_{A \sim P_{\text{data}}(A)}[(D_B(G_{AB}(R_A)))^2].
\]

where, \( \mathcal{L}_{\text{LSGAN}_B} \) is the Adversarial loss in domain \( B \). In the similar fashion, the \( \text{Synthesized Image} \) \((\text{Syn}_A)\) in domain \( A \) is generated from the Real Image \((R_B)\) of domain \( B \) by using the generator network \( G_{BA} : B \rightarrow A \). The Real Image \((R_A)\) and the \( \text{Synthesized Image} \) \((\text{Syn}_A)\) in domain \( A \) are differentiated by using the discriminator network \( D_A \). It is written as,

\[
\mathcal{L}_{\text{LSGAN}_A}(G_{BA}, D_A, B, A) = E_{A \sim P_{\text{data}}(A)}[(D_A(R_A) - 1)^2] + E_{B \sim P_{\text{data}}(B)}[(D_A(G_{BA}(R_B)))^2].
\]

where, \( \mathcal{L}_{\text{LSGAN}_A} \) is the Adversarial loss in domain \( A \). Adversarial loss is used to learn the distributions of the input data during training and to produce the real looking images in testing with the help of that learned distribution. The Adversarial loss tries to eliminate the problem of outputting blurred images, some artifacts are still present.

2) \textbf{Synthesized Loss:} Image-to-image transformation is not only aimed to transform the input image from source domain to target domain, but also to generate the output image as much as close to the original image in the target domain. To fulfill the later one the Synthesized loss is introduced in [42]. It computes the \( L_1 \) loss in domain \( A \) between the Real Image \((R_A)\) and the \( \text{Synthesized Image} \) \((\text{Syn}_A)\) and gives as,

\[
\mathcal{L}_{\text{Syn}_A} = \| R_A - \text{Syn}_A \|_1 = \| R_A - G_{BA}(R_B) \|_1 \quad (7)
\]

where, \( \mathcal{L}_{\text{Syn}_A} \) is the Synthesized loss in domain \( A \), \( R_A \) and \( \text{Syn}_A \) are the Real and Synthesized images in domain \( A \).

In the similar fashion, the \( L_1 \) loss in domain \( B \) between the Real Image \((R_B)\) and the \( \text{Synthesized Image} \) \((\text{Syn}_B)\) is computed as the Synthesized loss and given as,

\[
\mathcal{L}_{\text{Syn}_B} = \| R_B - \text{Syn}_B \|_1 = \| R_B - G_{AB}(R_A) \|_1 \quad (8)
\]

where, \( \mathcal{L}_{\text{Syn}_B} \) is the Synthesized loss in domain \( B \), \( R_B \) and \( \text{Syn}_B \) are the Real and Synthesized images in domain \( B \). Synthesized loss helps to generate the fake output samples closer to the real samples in the target domain.

3) \textbf{Cycle-consistency Loss:} To reduce the discrepancy between the two different domains, the Cycle-consistency loss is introduced in [27]. The \( L_1 \) loss in domain \( A \) between the Real Image \((R_A)\) and the Cycled Image \((\text{Cyc}_A)\) is computed as the Cycle-consistency loss and defined as,

\[
\mathcal{L}_{\text{Cyc}_A} = \| R_A - \text{Cyc}_A \|_1 = \| R_A - G_{BA}(G_{AB}(R_A)) \|_1 \quad (9)
\]

where, \( \mathcal{L}_{\text{Cyc}_A} \) is Cycle-consistency loss in domain \( A \), \( R_A \) and \( \text{Cyc}_A \) are the Real and Cycled images in domain \( A \). In the similar fashion, the \( L_1 \) loss in domain \( B \) between the Real Image \((R_B)\) and the Cycled Image \((\text{Cyc}_B)\) is computed as the Cycle-consistency loss and defined as,

\[
\mathcal{L}_{\text{Cyc}_B} = \| R_B - \text{Cyc}_B \|_1 = \| R_B - G_{AB}(G_{BA}(R_B)) \|_1 \quad (10)
\]

where, \( \mathcal{L}_{\text{Cyc}_B} \) is Cycle-consistency loss in domain \( B \), \( R_B \) and \( \text{Cyc}_B \) are the Real and Cycled images in domain \( B \). The Cycle-consistency losses, i.e., \( \mathcal{L}_{\text{Cyc}_A} \) and \( \mathcal{L}_{\text{Cyc}_B} \) used in the objective function act as both forward and backward consistencies. These two Cycle-consistency losses are also included in the objective function of the proposed CDGAN method. The scope of different mapping functions for larger networks is reduced by these losses. They also act as the regularizer for learning the network parameters.

4) \textbf{Cyclic-Synthesized Loss:} The Cyclic-Synthesized loss introduced in CSGAN [43], which is computed as the \( L_1 \) loss between the \( \text{Synthesized Image} \) \((\text{Syn}_A)\) and the \( \text{Cycled Image} \) \((\text{Cyc}_A)\) in domain \( A \) and defined as,

\[
\mathcal{L}_{\text{CS}_A} = \| \text{Syn}_A - \text{Cyc}_A \|_1 = \| G_{BA}(R_B) - G_{BA}(G_{AB}(R_A)) \|_1 \quad (11)
\]

where, \( \mathcal{L}_{\text{CS}_A} \) is the Cyclic-Synthesized loss, \( \text{Syn}_A \) and \( \text{Cyc}_A \) are the Synthesized and Cycled images in domain \( A \).

Similarly, the \( L_1 \) loss in domain \( B \) between the \( \text{Synthesized Image} \) \((\text{Syn}_B)\) and the \( \text{Cycled Image} \) \((\text{Cyc}_B)\) is computed as the Cyclic-Synthesized loss and defined as,

\[
\mathcal{L}_{\text{CS}_B} = \| \text{Syn}_B - \text{Cyc}_B \|_1 = \| G_{AB}(R_A) - G_{AB}(G_{BA}(R_B)) \|_1 \quad (12)
\]

where, \( \mathcal{L}_{\text{CS}_B} \) is the Cyclic-Synthesized loss, \( \text{Syn}_B \) and \( \text{Cyc}_B \) are the Real and Cycled images in domain \( B \). These two Cyclic-Synthesized losses are also included in the objective function of the proposed CDGAN method. Although, all the above mentioned losses are included in the objective function of the proposed CDGAN method, still there is a lot of scope to improve the quality of the images generated and to remove the unwanted artifacts produced in the resulting images. To fulfill that scope, a new loss called Cyclic-Discriminative Adversarial loss is proposed in this paper to generate the high quality images with reduced artifacts.

5) \textbf{Proposed Cyclic-Discriminative Adversarial Loss:} The Cyclic-Discriminative Adversarial loss proposed in this paper is calculated as the adversarial loss between the Real Images \((R_A \text{ and } R_B)\) and the Cycled Images \((\text{Cyc}_A \text{ and } \text{Cyc}_B)\). The adversarial loss in domain \( A \) between the Cycled Image \((\text{Cyc}_A)\) and the Real Image \((R_A)\) by using the generator \( G_{BA} \) and the discriminator \( D_A \) is computed as the Cyclic-Discriminative adversarial loss in domain \( A \) and defined as,

\[
\mathcal{L}_{\text{CDGAN}_A}(G_{BA}, D_A, B, A) = E_{A \sim P_{\text{data}}(A)}[(D_A(R_A) - 1)^2] + E_{B \sim P_{\text{data}}(B)}[(D_A(G_{BA}(\text{Syn}_B)))^2]. \quad (13)
\]
TABLE I: Showing relationship between the six benchmark methods and the proposed CDGAN method in terms of losses.

| Methods       | $L_{\text{LSGAN}_A}$ | $L_{\text{LSGAN}_B}$ | $L_{\text{Syn}_A}$ | $L_{\text{Syn}_B}$ | $L_{\text{Cyc}_A}$ | $L_{\text{Cyc}_B}$ | $L_{\text{CS}_A}$ | $L_{\text{CS}_B}$ | $L_{\text{CDGAN}_A}$ | $L_{\text{CDGAN}_B}$ |
|---------------|------------------------|------------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| Pix2Pix [40]  | ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |
| **DualGAN** [26]| ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |
| CycleGAN [27] | ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |
| PS2GAN [42]   | ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |
| CSGAN [35]    | ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |
| CDGAN (Ours)  | ✓                      | ✓                      | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   | ✓                   |

where, $L_{\text{CDGAN}_A}$ is the Cyclic-Discriminative adversarial loss in domain $A$, $C_{\text{yc}_A} (= G_{AB}(\text{Syn}_B))$ and $R_A$ are the cycled and real image respectively.

Similarly, the adversarial in domain $B$ loss between the Cycled Image ($C_{\text{yc}_B}$) and the Real Image ($R_B$) by using the generator $G_{AB}$ and the discriminator $D_B$ is computed as the Cyclic-Discriminative adversarial loss in domain $B$ and defined as,

$$L_{\text{CDGAN}_B}(G_{AB}, D_B, A, B) = E_{B \sim \rho_{\text{data}(B)}}[(D_B(R_B) - 1)^2] + E_{A \sim \rho_{\text{data}(A)}}[(D_B(G_{AB}(\text{Syn}_A)))^2].$$

(14)

where, $L_{\text{CDGAN}_B}$ is the Cyclic-Discriminative adversarial loss in domain $B$, $C_{\text{yc}_B} (= G_{AB}(\text{Syn}_A))$ and $R_B$ are the cycled and real image respectively. Finally, we combine all the losses to have the CDGAN Objective function. The relationship between the proposed CDGAN method and six benchmark methods are shown in Table I for better understanding.

### B. CDGAN Objective Function

The CDGAN method final objective function combines existing Adversarial loss, Synthesized loss, Cycle-consistency loss and Cyclic-Synthesized loss along with the proposed Cyclic-Discriminative Adversarial loss as follows,

$$L(G_{AB}, G_{BA}, D_A, D_B) = L_{\text{LSGAN}_A} + L_{\text{LSGAN}_B} + \mu_A L_{\text{Syn}_A} + \mu_B L_{\text{Syn}_B} + \lambda_A L_{\text{Cyc}_A} + \lambda_B L_{\text{Cyc}_B} + \omega_A L_{\text{CS}_A} + \omega_B L_{\text{CS}_B} + L_{\text{CDGAN}_A} + L_{\text{CDGAN}_B},$$

(15)

where $L_{\text{CDGAN}_A}$ and $L_{\text{CDGAN}_B}$ are the proposed Cyclic-Discriminative Adversarial losses described in subsection II-A5, $L_{\text{LSGAN}_A}$ and $L_{\text{LSGAN}_B}$ are the adversarial losses, $L_{\text{Syn}_A}$ and $L_{\text{Syn}_B}$ are the Synthesized losses, $L_{\text{Cyc}_A}$ and $L_{\text{Cyc}_B}$ are the Cycle-consistency losses and $L_{\text{CS}_A}$ and $L_{\text{CS}_B}$ are the Cyclic-Synthesized losses explained in the subsections II-A1 and II-A2 respectively. The $\mu_A$, $\mu_B$, $\alpha_A$, $\alpha_B$, $\omega_A$ and $\omega_B$ are the weights for the different losses. The values of these weights are set empirically.

### III. EXPERIMENTAL SETUP

This section is devoted to describe the datasets used in the experiment with train and test partitions, evaluation metrics used to judge the performance and the training settings used to train the models. This section also describes the network architectures and baseline GAN models.

#### A. DataSets

For the experimentation we used the following three different baseline datasets meant for the image-to-image transformation task.

1) **CUHK Face Sketch Dataset**: The CUHK dataset consists of 188 students face photo-sketch image pairs. The cropped version of the data with the dimension $250 \times 200$ is used in this paper. Out of the total 188 images of the dataset, 100 images are used for the training and the remaining images are used for the testing.

2) **CMP Facades Dataset**: The CMP Facades dataset contains a total of 606 labels and corresponding facades image pairs. Out of the total 606 images of this dataset, 400 images are used for the training and the remaining images are used for the testing.

3) **RGB-NIR Scene Dataset**: The RGB-NIR dataset consists of 477 images taken from 9 different categories captured in both RGB and Near-infrared (NIR) domains. Out the total 477 images of this dataset, 387 images are used for the training and the remaining 90 images are used for the testing.

#### B. Training Information

The network is trained with the input images of fixed size $256 \times 256$, each image is resized from the arbitrary size of the dataset to the fixed size of $256 \times 256$. The network is initialized with the same setup in [40]. Both the generator and the discriminator networks are trained with batch size 1 from scratch to 200 epochs. Initially, the learning rate is set to 0.0002 for the first 100 epochs and linearly decaying down to 0 over the next 100 epochs. The weights of the network are initialized with the Gaussian distribution having mean 0 and standard deviation 0.02. The network is optimized with the Adam solver [45] having the momentum term $\beta_1$ as 0.5 instead of 0.9, because as per [46] the momentum term $\beta_1$ as 0.9 or higher values can cause to substandard network stabilization for the image-to-image transformation task. The values of the weight factors for the proposed CDGAN method, $\mu_A$ and $\mu_B$ are set to 15, $\lambda_A$ and $\lambda_B$ are set to 10 and $\omega_A$ and $\omega_B$ are set to 30 (see Equation 15). For the comparison methods, the values for the weight factors are taken from their source papers.

1. [http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html](http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html)
2. [http://cmp.felk.cvut.cz/ tylecrl/facade/](http://cmp.felk.cvut.cz/ tylecrl/facade/)
3. [https://ivrl.epfl.ch/research-2/research-downloads/supplementary_material-cvpr11-index.html](https://ivrl.epfl.ch/research-2/research-downloads/supplementary_material-cvpr11-index.html)
C. Network Architectures

Network architectures of the generator and the discriminator as shown in Fig. 3 and used in this paper are taken from [27]. Following is the 9 residual blocks used in the generator network:

\[ \text{C}_{7 \times 7} \text{S}_1 \text{f}, \text{C}_{3 \times 3} \text{S}_2 \text{f}, \text{C}_{3 \times 3} \text{S}_2 \text{f}, \text{RB}_\text{256} \times 9, \text{DC}_{3 \times 3} \text{S}_2 \text{f}, \text{DC}_{3 \times 3} \text{S}_2 \text{f}, \text{C}_{7 \times 7} \text{S}_1 \text{f}, \]

where, \( \text{C}_{7 \times 7} \text{S}_1 \text{f} \) represents a \( 7 \times 7 \) Convolutional layer with \( f \) filters and stride 1, \( \text{C}_{3 \times 3} \text{S}_2 \text{f} \) represents a \( 3 \times 3 \) Convolutional_InstanceNorm_ReLU layer with \( f \) filters and stride 2, \( \text{RB}_f \times n \) represents \( n \) residual blocks consist of two Convolutional layers with \( f \) filters for both layers, and \( \text{DC}_{3 \times 3} \text{S}_2 \text{f} \) represents \( 3 \times 3 \) DeConvolution_InstanceNorm_ReLU layer with \( f \) filters and stride \( \frac{1}{2} \).

For the discriminator, we use \( 70 \times 70 \) PatchGAN from [40]. The discriminator network consists of: \( \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f}, \text{C}_{4 \times 4} \text{S}_2 \text{f} \), where \( \text{C}_{4 \times 4} \text{S}_2 \text{f} \) represents a \( 4 \times 4 \) Convolutional_InstanceNorm_ReLU layer with \( f \) filters and stride 2, and \( \text{C}_{4 \times 4} - 1 \) represents a \( 4 \times 4 \) Convolutional layer with \( f \) filters and stride 1 to produce the final one-dimensional output. The activation function used in this work is the Leaky ReLU with 0.2 slope. The first Convolution layer does not include the InstanceNorm.

D. Evaluation Metrics

To better understand the improved performance of the proposed CDGAN method, both the quantitative and qualitative metrics are used in this paper. Most widely used image quality assessment metrics for image-to-image transformation like Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), and Structural Similarity Index (SSIM) [47] are used under quantitative evaluation. Learned Perceptual Image Patch Similarity (LPIPS) proposed in [48] is also used for calculating the perceptual similarity. The distance between the ground truth and the generated fake image is computed as the LPIPS score. The enhanced quality images generated by the proposed CDGAN method along with the six benchmark methods and ground truth are also compared in the result section.

E. Baseline Methods

We compare the proposed CDGAN model with six benchmark models, namely, GAN [33], Pix2Pix [40], DualGAN [26], CycleGAN [27], PS2MAN [42] and CSGAN [43] to demonstrate its significance. All the above mentioned comparisons are made in paired setting only.

1) GAN: The original vanilla GAN proposed in [33] is used to generate the new samples from the learned distribution function with the given noise vector. Whereas, the GAN used for comparison in this paper is implemented for image-to-image translation from the Pix2Pix [40] by removing the \( L_1 \) loss and keeping only the adversarial loss.

2) Pix2Pix: For this method, the code provided by the authors in Pix2Pix [40] is used for generating the result images with the same default settings.

3) DualGAN: For this method, the code provided by the authors in DualGAN [26] is used for generating the result images with the same default settings from the original code.

4) CycleGAN: For this method, the code provided by the authors in CycleGAN [27] is used for generating the result images with the same default settings.

5) PS2GAN: In this method, the code is implemented by adding the synthesized loss to the existing losses of CycleGAN [27] method. For a fair comparison of the state-of-the-art methods with proposed CDGAN method, the PS2MAN [42]

4https://github.com/phillipi/pix2pix
5https://github.com/duxingren14/DualGAN
6https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix
method originally proposed with multiple adversarial networks is modified to single adversarial networks, i.e., PS2GAN.

6) CSGAN: For this method, the code provided by the authors in CSGAN\cite{43} is used for generating the result images with the same default settings.

IV. RESULT ANALYSIS

This section is dedicated to analyze the results produced by the introduced CDGAN method. To better express the improved performance of the proposed CDGAN method, we consider both the quantitative and qualitative evaluation. The proposed CDGAN method is compared against the six benchmark methods, namely, GAN, Pix2Pix, DualGAN, CycleGAN, PS2GAN and CSGAN. We also perform an ablation study over the proposed cyclic-discriminative adversarial loss with DualGAN, CycleGAN, PS2GAN and CSGAN to investigate its suitability with existing losses.

A. Quantitative Evaluation

For the quantitative evaluation of the results, four baseline quantitative measures like SSIM, MSE, PSNR and LPIPS are used. The average scores of these four metrics are calculated for all the above mentioned methods. The results over CUHK, FACADES and RGB-NIR scene datasets are shown in Table II. The larger SSIM and PSNR scores and the smaller MSE and LPIPS scores indicate the generated images with better quality. The followings are the observations from the results of this experiment:

- The proposed CDGAN method over CUHK dataset achieves an improvement of 26.93%, 13.14%, 7.75%, 4.81%, 6.91% and 3.56% in terms of the SSIM metric and reduces the MSE at the rate of 12.57%, 7.82%, 3.02%, 7.41%, 4.32% and 2.17% as compared to GAN, Pix2Pix, DualGAN, CycleGAN, PS2GAN and CDGAN, respectively.
- For the FACADES dataset, the proposed CDGAN method achieves an improvement of 82.29%, 19.27%, 675.30%, 270.50%, 42.40% and 15.07% in terms of the SSIM metric and reduces the MSE at the rate of 2.16%, 0.42%, 3.29%, 2.64%, 0.84% and 2.14% as compared to GAN, Pix2Pix, DualGAN, CycleGAN, PS2GAN and CDGAN, respectively.
- The proposed CDGAN method exhibits an improvement of 30.84%, 8.95%, 5072.22%, 5.15%, 4.94% and 7.55% in the SSIM score, whereas shows the reduction in the MSE score by 5.01%, 3.49%, 8.44%, 1.71%, 1.06% and 2.19% as compared to GAN, Pix2Pix, DualGAN, CycleGAN, PS2GAN and CDGAN, respectively.
- The performance of DualGAN is very bad over FACADES and RGB-NIR scene datasets because, these tasks involve high semantics-based labeling. The similar behavior of DualGAN is also observed by its original authors\cite{26}.
- The PSNR and LPIPS measures over CUHK, FACADES and RGB-NIR scene datasets also show the reasonable improvement due to the proposed CDGAN method compared to other methods.

From the above mentioned comparisons over three different datasets, it is clearly understandable that the proposed CDGAN method generates more structurally similar, less pixels to pixel noise and perceptually real looking images with reduced artifacts as compared to the state-of-the-art approaches. Note

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|}
\hline
\textbf{Datasets} & \multicolumn{3}{c|}{\textbf{Metrics}} & \multicolumn{6}{c|}{\textbf{Methods}} \\
\cline{2-10}
 & GAN\cite{33} & Pix2Pix\cite{40} & DualGAN\cite{26} & CycleGAN\cite{27} & PS2GAN\cite{42} & CSGAN\cite{43} & CDGAN \\
\hline
CUHK & \textbf{SSIM} & 0.5398 & 0.6056 & 0.6359 & 0.6537 & 0.6409 & 0.6616 & \textbf{0.6852} \\
 & MSE & 94.8815 & 89.9954 & 85.5418 & 89.6019 & 86.7004 & 84.7971 & \textbf{82.9547} \\
 & PSNR & 28.3628 & 28.5989 & 28.8351 & 28.6351 & 28.7779 & 28.8693 & \textbf{28.9801} \\
 & LPIPS & 0.157 & 0.154 & 0.132 & 0.099 & 0.098 & 0.094 & \textbf{0.090} \\
\hline
FACADES & \textbf{SSIM} & 0.1378 & 0.2106 & 0.0324 & 0.0678 & 0.1764 & \textbf{0.2183} \\
 & MSE & 103.8049 & \textbf{101.9864} & 105.0175 & 104.3104 & 102.4183 & 103.7751 & \textbf{101.5533} \\
 & PSNR & 27.9706 & 28.0569 & 27.9187 & 27.9849 & 28.032 & \textbf{27.9715} \\
 & LPIPS & 0.252 & \textbf{0.216} & 0.259 & 0.248 & 0.221 & 0.22 & \textbf{0.215} \\
\hline
RGB-NIR & \textbf{SSIM} & 0.4788 & 0.575 & \textbf{−0.0126} & 0.5958 & 0.597 & 0.5825 & \textbf{0.6265} \\
 & MSE & 101.6426 & 100.0377 & 105.4514 & 98.2278 & \textbf{97.5769} & 98.704 & \textbf{96.5412} \\
 & PSNR & 28.072 & 28.1464 & 27.9019 & 28.2574 & 28.2692 & \textbf{28.2159} \\
 & LPIPS & 0.243 & 0.182 & 0.295 & 0.18 & \textbf{0.166} & 0.178 & \textbf{0.147} \\
\hline
\end{tabular}
\caption{The quantitative comparison of the results of the proposed CDGAN with different state-of-the-art methods trained on CUHK, FACADES and RGB-NIR Scene Datasets. The average scores for the SSIM, MSE, PSNR and LPIPS metrics are reported. Best results are highlighted in bold and second best results are shown in italic font.}
\end{table}
that the SSIM of DualGAN on RGB-NIR dataset is negative as SSIM uses the co-variance between two images which can be negative, if the similarity in images is low.

B. Qualitative Evaluation

In order to show the improved quality of the output images produced by the proposed CDGAN, we compare and show few sample image results generated by the CDGAN against six state-of-the art methods. These comparisons over CUHK, FACADES and RGB-NIR scene datasets are shown in Fig. 4, 5 and 6 respectively. The followings are the observations and analysis drawn from these qualitative results:

- From Fig. 4, it can be observed that the images generated by the CycleGAN, PS2GAN and CSGAN methods contain the reflections on the faces for the sample faces of...
CUHK dataset. Whereas, the proposed CDGAN method is able to eliminate this effect due to its increased discriminative capacity. Moreover, the facial attributes in generated faces such as eyes, hair and face structure, are generated without the artifacts using the proposed CDGAN method.

- The proposed CDGAN method generates the building images over FACADES dataset with enhanced textural detail information such as window and door sizes and shapes. As shown in Fig. 5, the buildings generated by the proposed CDGAN method consist more structural information. In the first row of Fig. 5, the building generated by the CDGAN method from the top left corner contains more window information compared to the remaining methods.

- The qualitative results over RGB-NIR scene dataset are illustrated in Fig 6. The generated RGB images using the proposed CDGAN method contain more semantic, depth aware and structure aware information as compared to the state-of-the-art methods. It can be seen in the first row of Fig. 6 that the proposed CDGAN method is able to generate the grass in green color at the bottom portion of the generated RGB image, where other methods fail. Similarly, it can be also seen in 2nd row of Fig. 6 that the proposed CDGAN method generates the tree image in front of the building, whereas the compared methods fail to generate. The possible reason for such improved performance is due to the discriminative ability of the proposed method for varying depths of the scene points. It confirms the structure and depth sensitivity of CDGAN method.

- As expected, DualGAN completely fails to produce high semantics labeling based image-to-image transformation tasks over FACADES (see Column 4 of Fig. 5) and RGB-NIR (see Column 4 of Fig. 6) scene datasets.

It is evident from the above qualitative results that the images generated by CDGAN method are more realistic and sharp with reduced artifacts as compared to the state-of-the-art GAN models of image-to-image transformation.

C. Ablation Study

In order to analyze the importance of the proposed Cyclic-Discriminative adversarial loss, We conduct an ablation study on losses. We also investigate its dependency on other loss functions such as Adversarial, Synthesized, Cycle-consistency and Cyclic-Synthesized losses. Basically, the Cyclic-Discriminative adversarial loss is added to DualGAN, CycleGAN, PS2GAN, and CSGAN and compared with original results of these methods. The comparison results over the CUHK, FACADES and RGB-NIR scene datasets are shown in Table III. We observe the following points:

- The proposed Cyclic-Discriminative adversarial loss when added to the DualGAN and CycleGAN have a mix of positive and negative impacts on the generated images as shown in the DualGAN+ and CycleGAN+ columns in the Table III. Note that the proposed Cyclic-Discriminative adversarial loss is computed between the Real Image and the Cycled Image. Because the DualGAN+ and CycleGAN+ does not use the synthesized losses, our cyclic-discriminator loss becomes more powerful in these frameworks. It may lead to situation where the generator is unable to fool the discriminator. Thus, the generator may stop learning after a while.
Discriminative Adversarial loss are highlighted in italic and best results produced by the proposed CDGAN method are highlighted in bold.

TABLE III: The quantitative results comparison between the proposed CDGAN method and different state-of-the art methods trained on CUHK, FACADES and RGB-NIR Scene Datasets. The average scores for the SSIM, MSE, PSNR and LPIPS scores are reported. The ‘+’ symbol represents the presence of Cyclic-Discriminative Adversarial loss. The results with Cyclic-Discriminative Adversarial loss are highlighted in italic and best results produced by the proposed CDGAN method are highlighted in bold.

| Datasets | Metrics | Methods | DualGAN | DualGAN+ | CycleGAN | CycleGAN+ | PS2GAN | PS2GAN+ | CSGAN | CSGAN+ | Our Method |
|----------|---------|---------|---------|----------|----------|----------|---------|---------|--------|--------|------------|
| CUHK     | SSIM    | 0.6359  | 0.6357  | 0.6537   | 0.6309   | 0.6409   | 0.6835  | 0.6616  | 0.6762 | 0.6852 |            |
|          | MSE     | 85.5418 | 88.1905 | 89.6019  | 87.2625  | 86.7004  | 83.6166 | 84.7971 | 83.7534 | 82.9547 |            |
|          | PSNR    | 28.8351 | 28.6351 | 28.7444  | 28.6932  | 28.7779  | 28.9465 | 28.8693 | 28.947  | 28.9801 |            |
|          | LPIPS   | 0.132   | 0.104   | 0.099    | 0.109    | 0.098    | 0.106   | 0.094   | 0.105   | 0.090   |            |
| FACADES  | SSIM    | 0.0324  | 0.0542  | 0.0678   | 0.0249   | 0.1764   | 0.0859  | 0.2183  | 0.2408  | 0.2512  |            |
|          | MSE     | 105.0175| 105.0602| 104.3104 | 105.0355 | 102.4183 | 104.4056| 103.775 | 101.4973| 101.5533|            |
|          | PSNR    | 27.9187 | 27.9172 | 27.9489  | 27.9185  | 28.032   | 27.9446 | 27.9715 | 28.0767 | 28.0761|            |
|          | LPIPS   | 0.259   | 0.251   | 0.248    | 0.264    | 0.221    | 0.234   | 0.22    | 0.214   | 0.215   |            |
| RGB-NIR  | SSIM    | −0.0126 | 0.4088  | 0.5958   | 0.5795   | 0.597    | 0.6154  | 0.5825  | 0.6129  | 0.6265  |            |
|          | MSE     | 105.4514| 102.8754| 98.2278  | 98.1801  | 97.5769  | 97.3519 | 98.704  | 96.8126 | 96.5412|            |
|          | PSNR    | 27.9109 | 28.0116 | 28.2574  | 28.2619  | 28.2692  | 28.2874 | 28.2159 | 28.2933 | 28.3083|            |
|          | LPIPS   | 0.295   | 0.221   | 0.18     | 0.189    | 0.166    | 0.174   | 0.178   | 0.152   | 0.147   |            |

- It is also observed that the proposed Cyclic-Discriminative adversarial loss is well suited with the PS2GAN and CSGAN methods as depicted in the Table III. The improved performance is due to a very tough minimax game between the powerful generator equipped with synthesized losses and powerful proposed discriminator. Due to this very competitive adversarial learning, the generator is able to produce very high quality images.
- It can be seen in the last column of Table III that the CDGAN still outperforms all other combinations of losses. It confirms the importance of the proposed Cyclic-Discriminative adversarial loss in conjunction with the existing loss functions such as Adversarial, Synthesized, Cycle-consistency and Cyclic-Synthesized losses.

This ablation study reveals that the proposed CDGAN method is able to achieve the improved performance when proposed Cyclic-Discriminative adversarial loss is used with synthesized losses.

V. CONCLUSION

In this paper an improved image-to-image transformation method called CDGAN is proposed. A new Cyclic-Discriminative adversarial loss is introduced to increase the adversarial learning complexity. The introduced Cyclic-Discriminative adversarial loss along with the existing losses are used in CDGAN method. Three different datasets namely CUHK, FACADES and RGB-NIR scene are used for the image-to-image transformation experiments. The experimental quantitative and qualitative results are compared against GAN models including GAN, Pix2Pix, DualGAN, CycleGAN, PS2GAN and CSGAN. It is observed that the proposed method outperforms all the compared methods over all datasets in terms of the different evaluation metrics such as SSIM, MSE, PSNR and LPIPS. The qualitative results also point out the improved generated images in terms of the more realistic, structure preserving, and reduced artifacts. It is also noticed that the proposed method deals better with the varying depths of the scene points. The ablation study over different losses reveals that the proposed loss is better suited with the synthesized losses as it increases the competitiveness between generator and discriminator to learn more semantic features. It is also noticed that the best performance is gained after combining all losses, including Adversarial loss, Synthesized loss, Cycle-consistency loss, Cyclic-Synthesized loss and Cyclic-Discriminative adversarial loss.

ACKNOWLEDGEMENT

We are grateful to NVIDIA Corporation for donating us the NVIDIA GeForce Titan X Pascal 12GB GPU which is used for this research.

REFERENCES

[1] S. Zhang, R. Ji, J. Hu, X. Lu, and X. Li, “Face sketch synthesis by multdomain adversarial learning,” IEEE transactions on neural networks and learning systems, vol. 30, no. 5, pp. 1419–1428, 2018.
[2] M. Zhu, J. Li, N. Wang, and X. Gao, “A deep collaborative framework for face photo–sketch synthesis,” IEEE transactions on neural networks and learning systems, vol. 30, no. 10, pp. 3096–3108, 2019.
[3] C. Peng, X. Gao, N. Wang, D. Tao, X. Li, and J. Li, “Multiple representations-based face sketch–photo synthesis,” IEEE transactions on neural networks and learning systems, vol. 27, no. 11, pp. 2201–2215, 2015.
[4] Z. Cheng, Q. Yang, and B. Sheng, “Deep colorization,” in IEEE International Conference on Computer Vision, 2015.
[5] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in European Conference on Computer Vision, 2016.
[6] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, “High-resolution image inpainting using multi-scale neural patch synthesis,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.
