GAN-based paired image generation with dedicated encoding streams and dynamic balancing

Mohamed Kas, Yassine Ruichek

GAN-based paired image generation with dedicated encoding streams and dynamic balancing

Mohamed Kas[1],Yassine Ruichek[1]
[1] UTBM, CIAD UMR 7533, F-90010 Belfort, France
mohamed.kas@utbm.fr,
yassine.ruichek@utbm.fr

Abstract - Computer vision society experienced the birth of new CNN architecture known as Generative Adversarial Networks (GANs), which can generate fake images similar to real ones. The widespread use of GANs leads the image-to-image translation strategy dealing with more diverse tasks that were treated using traditional CNNs, such as medical analysis and semantic segmentation. In this paper, we propose a generic GAN referred to as Multi Streams with Dynamic Balancing-based Conditional Generative Adversarial Network (MSDB-CGAN). The MSDB-CGAN serves more challenging applications, that require multi input images such as binocular depth estimation, efficiently through its dedicated input streams and automatic skip connections. Moreover, the proposed GAN analyzes the inputs according to the target image, then assigns dynamic weights to the input streams. To validate the proposed MSDB-CGAN, we targeted two challenging tasks: binocular depth estimation and human-pose translation. These applications present different inputs requirements and configurations. The reported quantitative and qualitative comparisons prove that the MSDB-CGAN significantly outperforms the existing GANs as well as traditional CNN-based architectures.

Keywords: Generative Adversarial Learning, Conditional image generation, Dynamic balancing, Multi-Streaming inputs, Depth estimation, Human pose translation.

I. INTRODUCTION

Generative Adversarial Networks (GANs) are taking growing parts in computer vision applications, regarding their outstanding performance in image translation and generation. These networks have been used in different applications such as medical analysis [1], driving semantic segmentation [2], database generation [3]. The GANs are taking their strengths from being self-supervised through the generator/discriminator architecture, relying only on paired input and ground truth samples and no pixel annotation is required as compared to classic CNN frameworks. Furthermore, the GANs are an enhanced extension of the Encoder Decoder-based image generation networks by incorporating a discriminator part that supervises the generator to produce images similar to the targeted ones. So far, the existing GAN frameworks are used for application requiring one image as input that has to be translated it into a target image. However, some applications request many input images to compute the target image, which cannot be achieved using existing state-of-the-art GANs that support only one input. Many researchers proposed to stack the input images into channels then feed them to a first constitutional layer adapted to the number of concatenated channels, but the rest of the network is configured to process the stacked images as one input. This strategy suffers from a major drawback that relies on the early features fusion, which dramatically reduces the discriminating power at the following layers. This approach has been adopted in many works such as multi-modal image analysis [4], and RGB-D semantic segmentation [5] where depth information is added as a 4th channel with the input image. In both cases, the stacked inputs are not homogeneous and do not share the same sensors source (camera, MRI. . .). Thus, the convolution layers will not be able of encoding relevant filter response.

In this paper, we propose a new GAN architecture that supports multi inputs with dedicated streams and dynamic weights balancing of the inputs. The proposed GAN is referred to as Multi Stream Dynamic Balancing Conditional GAN (MSDB-CGAN). Moreover, we propose a new encoder decoder scheme based on a hybrid implementation of the U-Net and ResNet based generators. The proposed MSDB-CGAN includes an enhanced generative adversarial loss that considers visual quality of the produced image compared to the target one by calculating the similarity structure in addition to the L1 loss originally included in the GAN. For the discriminator part, we adopted pixel architecture since it delivers a good Fake/Real classification performance and forces the generator network to produce real looking images that are similar to the target. The dynamic weights balancing of the input streams makes our proposed GAN able to differentiate a main input image from an attention-based one. The attention-based input maps are adopted

to make the generator focusing on particular regions to be translated. Hence, the GAN should assign low weights to the attention features and high ones to the main inputs, since a set of their visual features will be included in the target image. The proposed MSDB-CGAN is a generic architecture that supports as many inputs as the application requires. To validate our proposed GAN, we targeted two different applications with different inputs challenges. The first application is depth estimation, which requires the left and right views to calculate the depth matrix. Estimating depth information from images is one of the basic and important tasks in computer vision, which is widely used in many applications such as SLAM [6]. The left and right images share the same level of information, and both of them are main inputs. Therefore, they should have similar streaming weights. Moreover, the output depth map is entirely different from the input's visual space, which will be a challenge for the proposed MSDB-CGAN as none of the input visual features will be included on the generated depth map. The second application is human pose generation based on two inputs : the main one is the actual pose of the person and the second is a patch of the desired pose, which can be seen as an attention mask. Therefore and through the dynamic weighing, the proposed MSDB-CGAN can translate the person pose according to the attention map while preserving the textural information related to the person clothes and skin.

To provide the readers and field interested researchers a better reading experiences, this paper is organized as follows. Section II introduces the proposed MSDB-CGAN and highlights in details its generator and discriminator architectures, in addition to the computation of the adversarial loss. Section III provides com- prehensive experiments on the four considered applications, that are very challenging and widely investigated by the state- of-the-art. The evaluation covers quantitative and qualitative assessments. The last section reviews some points that can be concluded from our work and presents some future works.

II. Multi Streams with Dynamic Balancing-based GAN

In this paper, we propose a new GAN referred to as Multi Stream Dynamically Balanced Conditional GAN (MSDB- CGAN). It works on encoding the features of different input images in order to generate a target image, which is judged via enhanced adversarial loss including the basic L1 loss and structure similarity-based one. The dynamic weighting of the streams makes the proposed GAN generic to any kind of input images that can be categorized into main input or attention- based one. This section is divided into three subsections to explain in-depth the overall architecture of the proposed GAN, the configurations of the generator and discriminator networks, and the adversarial loss computation.

A. Overall MSDB-CGAN architecture

The overall architecture of the proposed multi inputs image generation based on GAN is illustrated in Figure 1. The system processes each input image through a dedicated encoding stream in order to compute relevant features that allow a good generation at the decoding stage. Each stream includes a down sampling subnetwork and a residual-based feature extractor one. Then, all the streams are aggregated via a weighting layer that computes the weighted co-variance of the input image's variance to detect whether the input image presents big amount of data, in such case it is considered as a main input, or it presents less amount of data, in such case it is considered as an attention-based input. Afterwards, the aggregated features are fed to the decoder that is a set of up sampling layers to reconstruct the target image. The aforementioned components are coded as one network to form the generator part of the proposed MSDB-CGAN. Once the target image is generated, it is evaluated by the discriminator network to check if it looks like the target image (real image) or no (fake image). The adopted discriminator is a pixel-wise classification network and the decision on the image is the aggregation overall the pixels, which helps the generator to produce images more close to the target ones. Hence, the number of dedicated streams can be adjusted according to the application needs while granting an efficient feature extraction of each input image.

B. Generator and Discriminator networks configuration

The generator network of a GAN is an autoencoder architecture. The widely used ones in the state-of-the-art GANs are U-Net256, U-Net128, ResNet-6Blocks, and ResNet-9Blocks. The U-Net network was proposed in [9] for medical imaging purpose. The concept is based on supplementing down sampling (encoding) convolutional network by symmetric layers (decoding) where pooling operations are replaced by up sampling. Hence, these layers increase the resolution of the encoded image with more precision thanks to the skip connections. Moreover, in U-Net there are numerous feature channels in the up sampling part, which allow the network to propagate context information to higher resolution layers. As a consequence, the expansive path is more or less symmetric to the encoding part, and yields a U-shaped architecture. To predict the pixels in the border region of the image, the missing context is extrapolated by mirroring the input image. All the U-Net variants share the same pipeline, the only difference is the supported size of the images that is controlled by the amount of GPU memory and most of the works adopt 256 and 128 resolutions. On the other hand, a ResNet-based generator adopts residual blocks to compute relevant features from the input image, exploring the distinguishing power of the original ResNet [10]. Hence, the input image is down- sampled generally 2 or 3 times then the resulted convoluted feature maps are further processed by the residual subnetwork referred to as ResNet-Block. Afterwords, the output of the ResNet-Blocks is up

sampled to reach the specified size of the generator output. [8] reported that 6 and 9 blocks provide good generation. Furthermore, high number of blocks risks the gradient vanishing in addition to high computational cost. Inspired by the U-Net and ResNet-based autoencoders, we propose in this paper a new generator within the MSDB- CGAN that extends the autoencoder architecture to support multi streaming regarding the set of input images [I_0, ..., I_{n-1}]. As illustrated in Figure 1 each stream S_i has 3 down sampling blocks B^{DS} that produce a total set of 256 filters. These filters are fed to further feature extraction residual blocks B^{RFT} for more encoding. We adopted 5 blocks that experimentally proved to be efficient and discriminating. After encoding each input image, the feature aggregation phase is performed by computing the variance weight σ^C of each image I_i averaged on all the inputs. Therefore, the images with fewer pixels variations will have low weights and those that present rich data will have prominent weights. Therefore, the weight of each input image can be computed as formulated in Eq 1. The final aggregated feature set ζ is the concatenation of the weighted features $\eta_{\{n\}}$ of the n streams (cf. Eq 2 and 3).

$$w_i = \frac{\sigma_i^C}{\sum_{i=0}^{n-1} \sigma_i^C} \qquad (1)$$

$$\zeta = \Xi_{i=0}^{n-1}(w_i . \eta_i) \qquad (2)$$

where

$$\eta_i = S_i(I_i^C) \qquad (3)$$

Ξ is the depth concatenation operation performed on the set of extracted features $\eta_{\{n\}}$ from the n inputs, that are the condition images $I_{\{n\}}^C$. Note that, each stream S_i has its dedicated convolution weights and there is no sharing. This advantage guarantees an adequate feature extraction regarding the stream input. Afterwards, we proceed to the upscaling block B^{US} that reconstructs the target image I^g from the aggregated features ζ (cf. Eq 4). The upscaling block B^{US} is composed of 3 transposed convolutions that produce back the target input image with the same size as the condition images $I_{\{n\}}^C$.

$$I^g = U(\zeta) \qquad (4)$$

Moreover, we propose to use the skip connections between the encoding streams $S_{\{n\}}$ and the decoding one. The skip connections are used to recopy some visual features during the downsampling from one or more condition input images to be included in the generated upsampling layers. Hence, these connections help to have better visual quality, more sharpness and less blur. However, the challenge in our multi streams' implementation relies on figuring out a way to link all the visual features to the decoder block. The first scenario is to make these skip connections as a user enabled feature, depending on the

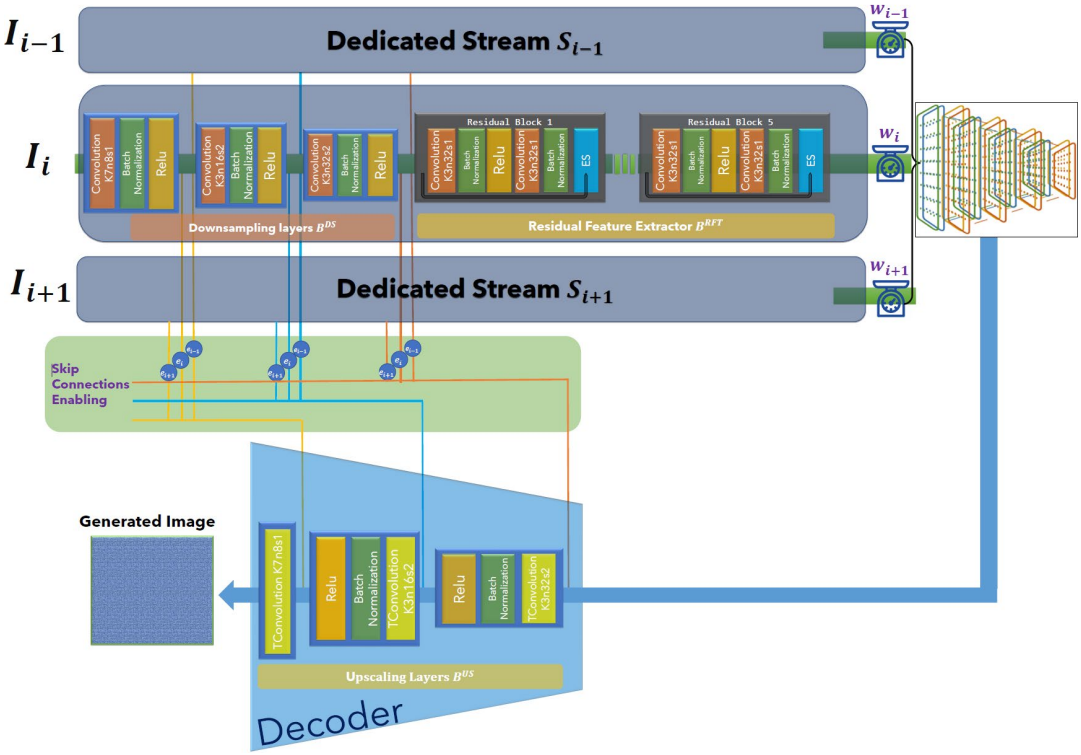

Fig. 1: Multi Inputs proposed generator configuration

application. For example, if we consider the depth estimation from stereo vision, the depth map has no visual features from the input images, and hence no skip connections will be needed. The second scenario, which we propose in our multi streams GAN, is to compare each input image with the corresponding ground truth image. The comparison is performed based on the variance and the tanh activation function at the generator network design (before proceeding to the training). Each input image $\boldsymbol{I_i^C}$ has a boolean variable $\boldsymbol{e_i}$ that enables or disables the skip connection between the stream $\boldsymbol{S_i}$ and the decoder. The boolean variable $\boldsymbol{e_i}$ is defined in Eq 5. If the variance $\boldsymbol{\sigma_i^C}$ of input image $\boldsymbol{I_i^C}$ is close to the target one $\boldsymbol{\sigma^{GT}}$, the value of $\boldsymbol{e_i}$ will be close to 1, and it will be close to 0 in the other case.

$$\boldsymbol{e_i = 1 - \left|\tanh\left(\sigma_i^C - \sigma^{GT}\right)\right|} \qquad (5)$$

After identifying the input images that will be connected to the decoder, we define the link $\boldsymbol{l_j}$ that is the skip connection between $\boldsymbol{j^{th}}$ downsampling layers of the encoding streams and the $\boldsymbol{j^{th}}$ upsampling layer of the decoder. $\boldsymbol{l_j}$ is defined as follows:

$$\boldsymbol{l_j = \sum_{i=0}^{n-1} e_i \,.\, S_{i,j}\left(I_i^C\right) \quad / \quad j = 0, 1, 2} \qquad (6)$$

where $\boldsymbol{i}$ denotes the number of input images and $\boldsymbol{j}$ for the downsampling layers on each stream that is fixed to 3. Therefore, $\boldsymbol{S_{i,j}\left(I_i^C\right)}$ is the downsampled filters of input image $\boldsymbol{I_i^C}$ extracted from $\boldsymbol{j^{th}}$ layer. Hence, all the components of our proposed generator have been defined to support multi streaming and the generated image will be fed to the discriminator network to judge its correspondence with the ground truth.

The discriminator is charged to supervise the generator network and differentiates the real images from the fake ones produced by the generator. Therefore, the discriminator is a binary pixel classification network including few convolution layers. The discriminator model adopted in the proposed MSDB-CGAN is implemented as Pixel discriminator. The Pixel discriminator assigns a binary label for each pixel in the generated / ground truth images. In general, the discriminator has a $\boldsymbol{N \times N}$ classification network that is processed convolutionally across the image to calculate the loss of each patch (non-overlapping blocks / pixel wise in case $\boldsymbol{N = 1}$), then all the responses are averaged to provide the overall loss, which is considered to update the networks weights through the optimizer.

C. MSDB-CGAN Loss

Similar to traditional CNN autoencoder, the GAN is trained via the back-propagation mechanism that relies on computing the loss of the network at each mini batch (iteration), and then, optimizes the network parameters. Referring to the original work , the generator tries to fool the discriminator by producing fake images looking like the real ones, while the discriminator seeks to correctly differentiate the fake from the real images. This process is a min-max game as Eq 7 formulates, where the generator tries to minimize the loss of produced frontal image detected as fake by the discriminator, which maximizes its performance to differentiate real from fake.

$$\boldsymbol{G^* = E_r\left|log\left(\mathrm{D}(I^{GT})\right)\right| + E_f\left|log\left(1 - \mathrm{D}\left(\mathrm{G}(I_{\{n\}}^C)\right)\right)\right|} \qquad (7)$$

where $\boldsymbol{\mathrm{D}(I^{GT})}$ refers to the discriminator's estimation of the probability that the ground truth image $\boldsymbol{I^{GT}}$ is classified as real. $\boldsymbol{E_r}$ is the expected value over all real data instances (ground truths). $\boldsymbol{\mathrm{G}\left(I_{\{n\}}^C\right)}$ is the generator's output from the given $\boldsymbol{n}$ input condition images. Therefore, $\boldsymbol{\mathrm{D}\left(\mathrm{G}\left(I_{\{n\}}^C\right)\right)}$ is the probability of detecting the generated images as fake instances. Hence, $\boldsymbol{E_f}$ is the expected value over all mini-batch inputs to the generator. While the discriminator is trained, it classifies both the real data and the fake data from the generator. It penalizes itself for misclassifying a real instance as fake, or a fake instance (created by the generator) as real, by maximizing the loss function (cf. Eq 8).

$$\boldsymbol{\nabla_{\mathrm{D}} \frac{1}{m} \sum_{k=1}^{m} \left[log\left(\mathrm{D}\left(I_k^{GT}\right)\right) + log\left(1 - \mathrm{D}\left(\mathrm{G}\left(I_{\{n\},k}^C\right)\right)\right)\right]} \qquad (8)$$

where $\boldsymbol{m}$ is the mini-batch size and $\boldsymbol{I_{\{n\},k}^C}$ are the $\boldsymbol{n}$ input images corresponding to $\boldsymbol{k^{th}}$ ground truth image from the selected mini-batch.

On the other hand, the generator output goes through the discriminator and gets classified as either "Real" or "Fake" based on the ability of the discriminator training. The generator loss is basically calculated based on the discriminator's classification of the produced image, it gets rewarded if it successfully fools the discriminator, and gets penalized otherwise. Moreover, extra loss functions can be considered to optimize the generator weights. In our work, we used the L1 and SSIM-based losses to help the generator to produce less blurry images. Structure Similarity (SSIM) measure is a perceptual metric that quantifies image quality degradation caused by processing such as data compression or by losses in data reproduction in case of GAN-based image generation. SSIM incorporates important structural information (luminance and contrast), meaning that the nearby pixels have strong interdependencies and carry information about the structure of the objects in the visual scene. Luminance tends to be less visible in bright regions, while contrast becomes less visible where there is significant activity in the image. SSIM ranges from 0 to 1, higher the better. Therefore, the loss used to update the generator weights is defined as follows:

$$\nabla_G \frac{1}{m}\sum_{k=1}^{m}[\,log\left(1-D\left(G\left(I^C_{\{n\},k}\right)\right)\right)+\lambda\times L_1\left(I^{GT}_k, G\left(I^C_{\{n\},k}\right)\right) + L_S\left(I^{GT}_k, G\left(I^C_{\{n\},k}\right)\right)] \quad (9)$$

where

$$L_1\left(I^{GT}_k, G\left(I^C_{\{n\},k}\right)\right) = \left|I^{GT}_k - G\left(I^C_{\{n\},k}\right)\right|_1 \quad (10)$$

and

$$L_S\left(I^{GT}_k, G\left(I^C_{\{n\},k}\right)\right) = \frac{1}{SSIM\left(I^{GT}_k, G\left(I^C_{\{n\},k}\right)\right)} \quad (11)$$

III. Experimental analysis

In this paper, we proposed a new GAN, referred to as MSDB-CGAN, that supports multi inputs-conditional image generation with dedicated streams. In order to prove the performance of the MSDB-CGAN, we targeted two challenging applications that require multi inputs.

A. Depth Estimation

The state-of-the-art works devoted to solve depth estimation from camera sensor adopted the Cityscapes benchmark. It provides comprehensive stereo runs and accurate depth annotation with clear Train/Test split to guarantee a fair comparison with the state-of-the-art. The train set includes 3,475 stereo images with the corresponding depth ground truth and 1,525 samples devoted for testing. We down sampled the resolution of the images to $\mathbf{512 \times 512}$ keeping the same aspect ratio as the original one using zero padding. To quantitatively evaluate the predicted depth maps using the proposed MSDB-CGAN, we calculate several standard evaluation metrics used in previous works . We adopted four error-based metrics that are the mean relative error (Rel), the squared relative error (Sq Rel) the root mean squared error (RMSE), and the mean log 10 error (logRMSE). We also used one accuracy-based metric ($\mathbf{A}^{\tau}$) with threshold τ.

Table I illustrates the achieved results on Cityscapes' depth test set based on the presented metrics along with the results reported in well-known state-of-the-art works. It can be inferred from Table I that the MSDB-CGAN outperformed the state-of-the-art by guarantying low errors and high accuracy. Furthermore, the accuracy results prove that the predicted depth is very close to the ground truth, that are generally calculated using classic depth methods with calibration parameters. Therefore, the two streams

TABLE I: Depth estimation quantitative results on CityScapes

Method	Rel	Sq Rel	RMSE	logRMSE	$\tau = 1.25$	$\tau = 1.25^2$	$\tau = 1.25^3$
MSDB-CGAN[b]	0.212	3.350	4.165	0.298	0.804	0.930	0.954
CRF-DGAN [16][b]	0.411	5.985	-	0.403	0.756	0.897	0.953
Laina et al. [17][m]	0.257	4.238	7.273	0.448	0.765	0.893	0.940
Zhang et al. [18][m]	0.234	3.776	7.104	0.416	0.776	0.903	0.949
Pad-net [19][m]	0.246	4.060	7.117	0.428	0.786	0.905	0.945
SDC-Depth [20][m]	0.227	3.800	6.917	0.414	0.801	0.913	0.950
Pilzer et al. [21][b]	0.440	6.036	5.443	0.398	0.730	0.887	0.944

^{m} : Monocular depth estimation, ^{b} : Binocular depth estimation.

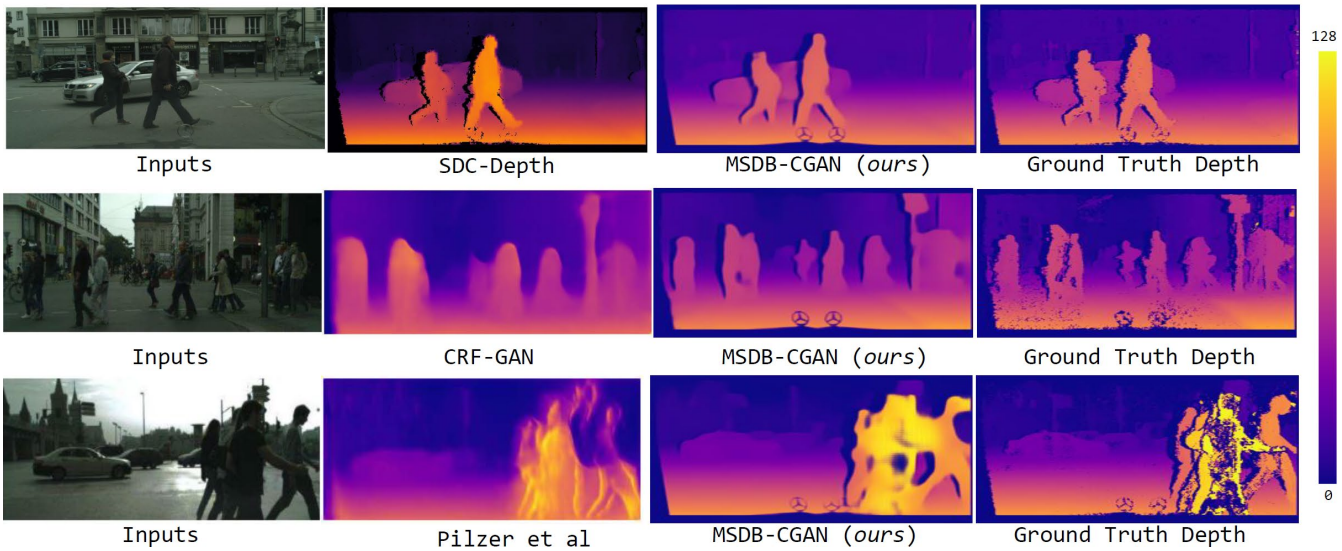

Fig. 2: Qualitative comparison of CityScapes Depth against SDC-Depth [20], CRF-GAN [16], and Pilzer et al. [21]

dedicated for the stereo images managed to encode the feature sets efficiently, and then, estimate with high accuracy the depth of each pixel from the mounted stereo camera. In terms of accuracy, we are the only to reach 93% considering $\boldsymbol{\tau = 1.25^2}$ while the state-of-the-art works were limited to 91.3% . Moreover, we managed to reduce the RMSE error by 25% of the best one from the state-of-the-art, which is 5.443 vs 4.165 by the MSDB-CGAN. Furthermore, Figure 2 presents a qualitative comparison of the predicted depth map images against the state-of-the-art available ones. The conclusions inferred from the quantitative comparison are once again confirmed through the qualitative comparison. If we consider the outputs of (first row), the depth of the two pedestrians is not accurate since they are very close and almost at the same distance from the camera, but the color intensity shows that the depth value of the male person is very higher than the one from the female. On the other hand, the produced MSDB-CGAN depth output of that stereo image is very close to the ground truth, which corresponds to accurate depth estimation. Compared to , our MSDB-CGAN predicted a more sharp and accurate depth map for the stereo images representing crossing pedestrians (second row), while their proposed model, which is GAN-based, generated a square block of depth for each person, which means the presence of false depth values. In addition, the third row of Figure 2 presents a peer comparison against the approach in , which is based on cyclic adversarial learning. It can be easily spotted that our predicted depth map is much more clear and similar to the ground truth, whereas the output of lacks of sharpness and the depth values within each object present artifacts and shades.

It can be concluded from the above statements that the MSDB-CGAN through the left and right dedicated streams managed to encode the visual features to extract the accurate depth information.

TABLE II: Quantitative IS and SSIM results on DeepFashion-based pose translation benchmark

Method	SSIM	IS
MSDB-CGAN	0.837	3.721
Deformable GAN [23]	0.756	3.439
Disentangled PG [24]	0.614	3.228
bFT [25]	0.767	3.22
Progressive Pose Attention [26]	0.773	3.209
RATE [27]	0.774	3.125
PG Squared [28]	0.762	3.090

B. Human pose translation

To evaluate our proposed MSDB-CGAN on human pose translation application, we considered DeepFashion benchmark , which contains about 50K images of fashion models in texture-rich clothes under three poses : Front, Back, and Side along with their corresponding masks. The images are in $\boldsymbol{512 \times 512}$ resolution and contain clean background. After checking the whole dataset, we found that it provides 263k person-related paired pose/mask samples. Therefore, we could gather 23k translations covering the three possible directions (front <=> back, front <=> side, side <=> back), where 18k are used to train and 5k for evaluation. The MSDB-CGAN in this case is configured to have two inputs, one is the actual pose and the second is the mask of the target pose, which is this time an attention-based input unlike the previous application of depth estimation where the two input images were primary ones. Moreover, the skip connections will consider only the actual pose input and not the target pose input (mask), as the latter doesn't contain any textural features to be recopied into the target image. The evaluation is performed by computing the Structure Similarity (SSIM) and Inception Score (IS) metrics as reported in the state-of-the-art works.

Fig. 3: Qualitative comparison on pose translation from DeepFashion database

The recorded metrics for this experiment are listed in Table II, that also includes the literature ones published in well indexed journals. The MSDB-CGAN scored the highest SSIM of 83.69% while the rest models are stuck at 78%. The superiority of our proposed GAN was also confirmed on the IS score by reaching 3.7209 with a clear gap of 0.3 above the state-of-the-art performance. The SSIM and IS metrics checking covers both texture and shape correspondence between the generated images and the ground truths ones. Hence, the high scores achieved prove that the MSDB-CGAN managed to encode the texture presented in the actual pose input and apply it to the target pose, which is the second input condition. This statement can be verified on the qualitative assessment presented in Figure 3. Speaking of the generated pose of the male (row one), none of the state-of-the-art approaches could generate a complete pose, neither predicting accurate clothes texture as well as the hairstyle and hands position. The MSDB-CGAN prediction was very close to the ground truth with green color issues, that can be enhanced by post-processing, and the clearest face details compared to the other approaches. The same remarks are valid on the women output (second row), we got the clearest and accurate pose translation as compared to the state-of-the-art generation, that couldn't maintain the color structure of the clothes and merge between them. The MSDB-CGAN proved on the DeepFashion dataset that it's a generic framework, as it managed to translate from various poses without being trained independently on each pose thanks to the dedicated streams. Furthermore, the colors contained in the target mask input did not affect the generation of the target pose, which is a straight benefit of the automatic skip connection enabling proposed in this paper.

C. Implementation and execution

The MSDB-CGAN is developed using the Python3.6 programming language along with the PyTorch 1.7 Neural Network Libraries with CUDA GPU Toolkit 11. The training is based on Adam optimizer with a learning rate of 0.0005 for all the experiments, and performed on Alienware Aurora R11 i9-10900KF Dual RTX2080Ti (22 GB VRAM).

IV. Conclusion and Perspectives

In this paper, we proposed a new adversarial learning-based generator referred to as Multi Streams Dynamic Balancing Conditional GAN, that enables the image generation according to different input images considered as conditions. Through a comprehensive state-of-the-art approaches review, we highlighted how the traditional GANs have been adopted to fulfill the applications requiring more than one input image. Mainly, the depth concatenation is adopted to make the different conditions as one input, which is then processed to proceed with feed-forward computation. Therefore, the dedicated streaming of inputs is not yet explored and adopted to build GAN structures. The proposed MSDB-CGAN is a generic framework capable of analyzing the input images to assign dedicated weights to each of computed features before the decoding stage. Moreover, we included a thresholding process of the input images with the target one as mean of enabling the skip connections linking the pairs of down sampling and up sampling layers. The adversarial learning is based on a pixel-wise discriminator network that decides if the generated image looks like the target one or no, then penalizes the generator through the adversarial loss. We also included the L1 and SSIM losses into the objective function of the MSDB-CGAN for more generation enhancement and to avoid the blurry pixels which often occur when using residual feature extractors. Through a comprehensive evaluation on two challenging applications : Stereo depth and Human pose translation, we proved that the proposed MSDB-CGAN scheme outperformed many recent state-of-the-art methods and managed to generate accurate images that are very close to the ground truths. The application's choice was meant to highlight the flexibility of the MSDB-CGAN in terms of the number of input images and their weights (main input / attention-based one). However, the bottleneck of the MSDB-CGAN mainly relies on the increasing number of parameters according to the number of input images, which will require more training time and computation resources.

As future works, we intend to incorporate computation reduction techniques to reduce the number of parameters while preserving a good generation performance. Moreover, we believe that the MSDB-CGAN could be applied to other interesting applications such as Multi Modal Medical Analysis as well as Kalman Filtering and trajectory prediction. Finally, we will consider the evaluation of deep-based loss functions like the VGG one that can be trained to meet the requirements of a given image generation application.

Acknowledgment

The authors gratefully acknowledge the funding received from CNSRT-Maroc (Centre National de la Recherche Scientifique et Technique) and the French government (Eiffel scholarship).

References

[1] K. Armanious, C. Jiang, M. Fischer, T. Ku¨stner, T. Hepp, K. Nikolaou, S. Gatidis, and B. Yang, "Medgan: Medical image translation using gans," *Computerized Medical Imaging and Graphics*, vol. 79, p. 101684, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0895611119300990

[2] X. Zhang, X. Zhu, . X. Zhang, N. Zhang, P. Li, and L. Wang, "Seggan: Semantic segmentation with generative adversarial network," in *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, 2018, pp. 1–5.

[3] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5781–5790.

[4] J. Gao, P. Li, Z. Chen, and J. Zhang, "A survey on deep learning for multimodal data fusion," *Neural Computation*, vol. 32, no. 5, pp. 829–864, 2020.

[5] Y. Guo and T. Chen, "Semantic segmentation of rgbd images based on deep depth regression," *Pattern Recognition Letters*, vol. 109, pp. 55–64, 2018.

[6] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE robotics & automation magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[7] X. Wang and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *European conference on computer vision*. Springer, 2016, pp. 318–335.

[8] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] R. Xu, Z. Zhou, W. Zhang, and Y. Yu, "Face transfer with generative adversarial network," *arXiv preprint arXiv:1710.06090*, 2017.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672– 2680.

[13] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in neural information processing systems*, vol. 27, pp. 2366–2374, 2014.

[14] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 270– 279.

[15] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille, "Towards unified depth and semantic prediction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2800–2809.

[16] M. M. Puscas, D. Xu, A. Pilzer, and N. Sebe, "Structured coupled generative adversarial networks for unsupervised monocular depth esti- mation," in *2019 International Conference on 3D Vision (3DV)*. IEEE, 2019, pp. 18–26.

[17] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *2016 Fourth international conference on 3D vision (3DV)*. IEEE, 2016, pp. 239–248.

[18] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang, "Joint task-recursive learning for semantic segmentation and depth estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 235–251.

[19] D. Xu, W. Ouyang, X. Wang, and N. Sebe, "Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 675–684.

[20] L. Wang, J. Zhang, O. Wang, Z. Lin, and H. Lu, "Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 541–550.*

[21] *Dsfsdf A. Pilzer, D. Xu, M. Puscas, E. Ricci, and N. Sebe, "Unsupervised adversarial depth estimation using cycled generative networks," in 2018 International Conference on 3D Vision (3DV). IEEE, 2018, pp. 587– 595.*

[22] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[23] A. Siarohin, E. Sangineto, S. Lathuiliere, and N. Sebe, "Deformable gans for pose-based human image generation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3408–3416.

[24] L. Ma, Q. Sun, S. Georgoulis, L. Van Gool, B. Schiele, and M. Fritz, "Disentangled person image generation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 99– 108.

[25] B. AlBahar and J.-B. Huang, "Guided image-to-image translation with bi-directional feature transformation," in Proceedings of the IEEE Inter- national Conference on Computer Vision, 2019, pp. 9016–9025.

[26] Z. Zhu, T. Huang, B. Shi, M. Yu, B. Wang, and X. Bai, "Progressive pose attention transfer for person image generation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2347–2356.

[27] L. Yang, P. Wang, X. Zhang, S. Wang, Z. Gao, P. Ren, X. Xie, S. Ma, and W. Gao, "Region-adaptive texture enhancement for detailed person image synthesis," in 2020 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2020, pp. 1–6.

[28] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," in Advances in neural information processing systems, 2017, pp. 406–416.