View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Hosei University Repository

法政大学学術機関リポジトリ
HOSEI UNIVERSITY REPOSITORY

Recovery from Compositely Degraded Images Using Conditional Generative Adversarial Networks

著者	Huang Yaoyi
出版者	法政大学大学院情報科学研究科
journal or publication title	法政大学大学院紀要．情報科学研究科編
volume	14
page range	1-6
year	2019-03-31
URL	http://doi.org/10.15002/00021926

Recovery from Compositely Degraded Images Using Conditional Generative Adversarial Networks

Huang Yaoyi
Graduate School of Computer and Information Sciences
Hosei University
Tokyo 184-8584, Japan
yaoyi.huang.24@stu.hosei.ac.jp

***Abstract*—The recovery from degraded images is an important technology that is used widely in many fields, and various approaches have been studied. Generally, when the image degradation model is simple and its degree is light, for example, when the image is degraded by a single motion blur, it is easy to recover. But when the degradation is caused by mixture of different types of blurs, and when it is additionally affected by noise, then the recovery is far more difficult. To address this problem, we propose a solution based on the Conditional Generative Adversarial Networks (cGANs) which not only learns the mapping from input image to output image, but also learns a loss function to train this mapping. In this thesis, we report the evaluation results of our cGAN-based implementation using face data sets, and show that our approach significantly improves the performance for the compositely degraded images.**

Keywords:Compositely Degraded Images, Conditional Generative Adversarial Networks, Image Recovery;

I. INTRODUCTION

Recover degraded images always been a problem that has plagued the industry in image processing. The causes of image blurring can be very complicated. For example, camera shake, out of focus, high-speed movement of objects, fog, noise, its own resolution is not enough, and many other reasons. And all require an effective method to turn an unclear blurred image into a clear image. But now the exiting method to recover compositely degraded images are usually not satisfactory because many of them only have good effect on a specific single degraded image.

We explore the GAN in the conditions set in. As GAN learning data generation model, GAN (cGAN) generation model learning conditions. This makes the cGAN suitable for the task of image restoration of degraded images to the image, which we performed on the input image and generate the corresponding image output regulation.

GAN has been actively studied in the past two years, we put forward a lot of technology we use in this paper. Nevertheless, early paper focuses on the specific application, it is unclear how GAN image as an image to image conversion solution. And I found that conditional GANs can be have a good effective on the recovery of compositely degraded images since other method only works well on the single degraded or light degraded like only motion blur.

Supervisor : Prof. Kaoru Uchida

And in this paper, I use different methods to put some blur or noise that makes image degrade. I combine the Motion Blur, Gaussian Blur, Bilateral Filter, Salt-and-Pepper Noise in different ways and different levels.And the ultimate goal for the research is to recover the compositely degraded images. We apply the Conditional Generative Adversarial Networks to solve that problem.

This thesis is organized as follows: in section I, we explain the background and what will do in this thesis. In section II we will explain about related works, like DeblurGans [1] , Image Deblurring, Generative adversarial networks. Section III introduces our approach to address the problem using the our cGAN-based solution method. In section IV we explain about the evaluation and its result, then concluded it in section V.

II. RELATED WORK

A. Image Deblurring

The on-uniform blur model's formulation usually be like this:

$$I_B = k(M) * I_S + N$$

The *IB* is a fuzzy image, *K (M)* is determined by the field *M* unknown blur kernel, *IS* is clearly a latent image, * denotes convolution, N additive noise. The fuzzy problem of the family can be split into two parts: blind and non blind person. In earlier work [2], many works mainly focus on non blind deblurring assumptions, fuzzy kernel *K (M)* is known to. Most of them rely on the classical L-Jackestion algorithm, Wiener or Tikhonov filter to perform deconvolution and obtain *IS* estimates. Usually fuzzy function is unknown, and the blind deblurring algorithm to estimate the clear image of *IS* potential *(M)* and fuzzy kernel *K*. Find the ambiguity function of each pixel is an ill posed problem, and most of the existing algorithms rely on heuristics, image statistics and a fuzzy source assumption. These methods by considering the fuzzy image uniformity to solve the ambiguity caused by camera jitter. First of all, based on camera motion induced blur kernel estimation, and then by performing the deconvolution to reverse the effect of. From the success of Fergus et al. [3] [7] [5] [8], many methods have been developed over the past ten years. Some methods based on iterative method of [8] [7], which improves the clear image estimation and each iteration of the nuclear motion through the use of a priori parameters model. Others use the fuzzy function of local linear hypothesis and simple heuristic method to estimate the unknown nuclear rapid. Others use the fuzzy function of local linear hypothesis and simple heuristic method to estimate the unknown nuclear rapid. This method is very fast, but they are suitable for small image.

Recently, White and others the [4] parameter geometric model of rotating speed camera exposure process based on the development of a new non uniform blind deblurring algorithm. The same Gupta et al. [9] fuzzy hypotheses only by the mobile camera caused by 3D. With the deep learning success over the past few years, there have been some based on convolutional neural network (CNN) approach. The sun and the others. [10] CNN Chakrabarti [11] estimate fuzzy kernel, prediction of complex Fu Liye coefficient of the nuclear motion in the Fu Liye space to perform non blind and Gong [12] using fuzzy, convolutional network complete moving motion estimation flow. All of these methods are used to estimate the unknown function fuzzy CNN. Recently, Noorozi and Nah by using the method of kernel free end to end, and the use of multi-scale CNN directly delete images. Ramakrishnan et al. We use [9] to perform blind image deblurring based on kernel cGAN solution framework and the combination of densely connected convolutional network. These methods can handle different fuzzy source.

B. Generative Adversarial Networks

The concept proposed by Goflliow is the definition of network confrontation between two matches in competition: network discriminator and generator. The noise generator receives the input and generates the sample. Receive actual and generated samples and try to distinguish them. The goal is to generate generators that are indistinguishable from real samples, so as to identify convincing samples to discriminate discriminator. Between *G* and *D*, game generator discriminator is a very small goal:

The concept of *GF* is proposed for the network, which defines two competitors' online games: the discriminator and the generator. The noise generator receives the input and generates the sample. Receive actual and generated samples and try to distinguish them. The target is a convincing sample deception discriminator generated by the generator and unable to distinguish between perceptual and actual samples. Between *G* and *D*, game generator discriminator is the smallest and largest goal:

$$\min_{G}\max_{D} \mathop{\mathrm{E}}_{x\ \mathrm{P}_r}[\log(D(x))] + \mathop{\mathrm{E}}_{\tilde{x}\ \mathrm{P}_g}[\log(D(\tilde{x}))]$$

where PR is the data distribution, *Pg* distribution model, *x = g (z), Z P (z) Z* is defined, the input from simple noise distribution. The ability of GANS to its production has good perceptual quality of the sample is known. However, as described by the improved training GANS technology, vanilla edition training suffers many problems, such as the collapse of patterns, the gradient of disappearance and so on. To minimize the value of the GAN function is equal to the minimum Jaskon Shanne divergence between data and model the distribution of ARJARSky on the [15]. The approximate GAN caused by the difficulties of training by JS divergence, and put forward the use of bulldozers *(Q, P)* W distance. The value function of WAN is to use KAN-RuBN dual construction:

$$\min_{G}\max_{D\in\mathrm{D}} \mathop{\mathrm{E}}_{x\ \mathrm{P}_r}[D(x)] + \mathop{\mathrm{E}}_{\tilde{x}\ \mathrm{P}_g}[D(\tilde{x})]$$

D is a set of Lipchit function, and *P* is the distribution model. The idea here is to approximate the critical value of *K - W (PR P THI),* where *K* is constant Lipchit , and *W (PR, P, Y)* is the distance of the accelerator. In the collection, discriminator network is known as the critics, which approximates the distance between two samples. In order to improve the Lipchit constraint in WGA-ARJOVSK [15], the weight limit is increased to *[-C, C]*. Gura Jenny, etc.. [6] proposed to increase the gradient penalty.

$$\lambda \mathop{\mathrm{E}}_{\tilde{x}\sim\mathrm{P}_{\tilde{x}}}[\|\nabla_{\tilde{x}} D(\tilde{x}\|_2 - 1)^2]$$

value functions serve as an alternative to forcing Lipschitz constraints. This method for generator architecture is robust, and almost do not need super parameter adjustment. This is essential for image deblurring, because it allows the use of a novel lightweight neural network architecture rather than a standard Deep ResNet architecture used to blur the image before.

C. Deblur GANs

DeblurGANs [1] is a single blind light motion. Through the application of network warfare (GAN), great progress has been made in the field of super resolution image and painting recently. GAN can maintain image details, create realistic image manifold solutions, and feel convincing. Recently, the image super-resolution and image generation network maps the inspiration of translation work to the image as a special case of DeblurGANs blurred image to image translation. DeblurGAN is a method of generating network and multi components based on the condition of loss function. Unlike previous works, DeblurGAN Wasserstein GAN gradient and perceived penalty are used. This encourages clear perception and image segmentation solutions, and uses traditional MSE (mean square error) or MAE (mean absolute error) to allow more detailed texture recovery details compared with the optimization target.

The target only restores the *IS* input as a clear image in the blurred image *IB*, so it does not provide information about the fuzzy kernel. After stripping, we completed the training of CNNG theta *G*, which we call generators. For each *IB*, the *IS* image is estimated. In addition, in the training phase, it introduced *D* and *D* training network to criticize theta two networks to fight.

The generator is similar to the Johnson CNN architecture for architectural style transfer tasks. It consists of two steps, nine block convolutions, residual blocks (ResBlocks) and two transposition convolutions. Each ResBlock is based on the volume layer instance standardization layer and ReLU activation combination. After adding probability 0.5 in the first volume layer of each ResBlock, loss regularization. In addition, DeblurGANS also uses the global connection named ResOut skip. CNN learning IB residual blurred image correction IR = IB + IR, so IS. The review network architecture is the same as that of PatchGAN. In addition to the final, all volumes are after the InstanceNorm layer and the LeakyReLU layer, including alpha = 0.2.

III.METHODOLGY

We study the conditions on the network as a general recovery hybrid solution to downgrade images. These networks not only learn to map from input images to output images, but also learn loss function mapping training. This makes it possible to use the same general method to lose very different formulations on traditional problems. GAN has been actively studying in the past two years, and many of the technologies we explored in this article were previously proposed. Nevertheless, the focus of early papers is on specific types of degradation. The main contribution of our method lies in various image quality, and the conditional GAN produces reasonable results.

GAN is a generative model, from random noise vector Z to output image y, G:z mapping to y. Instead, GAN G:{x x learn from observed images, random noise vectors Z to y, z}, y, map. The training generator G can not segment the actual image by the training side and the output D discriminator, and the discriminator can train the false detection generator as far as possible. The training plan is shown in Figure 1.

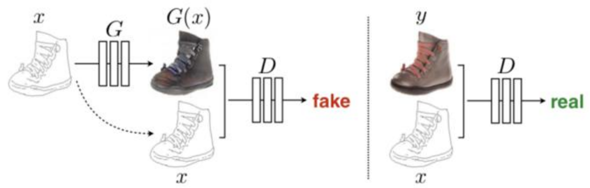

Figure 1:GAN maps the edges to the training conditions of the photos. In D Research (by pseudo discriminator generator synthesis) and real (edge, photo) classification between tuples. Learn to use the generator G discriminator. Unlike the unconditional GAN generator and the discriminator on the map, input the edge.[16]

A. Objective

The conditional GAN can objective be showed as flow

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_y[logD(y)] + \mathbb{E}_{x,z}[log(1-D(G(x,z)))]$$

G tries to minimize this goal, instead of trying to maximize the antagonism of D.

In order to test the importance of adjusting the discriminator, we compared a conditional variable that was not observed in the x discriminator:

$$\mathcal{L}_{GAN}(G,D) = \mathbb{E}_y[logD(y)] + \mathbb{E}_{x,z}[log(1-D(G(x,z)))]$$

Previous methods have found that GAN targets with a more conventional loss (such as L2 distance) mixtures are useful. The discriminator work remains unchanged, but the task is not only deception generator discriminator, but also close to the true meaning of L2 output on the ground. We also explored this option, using L1 distance instead of L2, because L1 is encouraged to reduce ambiguity.

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\| y - G(x,z) \|_1]$$

Our final goal is

$$G^* = arg \min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G)$$

Without Z, mapping from X to y networks can still be done, but output uncertainty will be generated, so it is impossible to match any distribution outside the delta function. GAN has acknowledged this in the past, and in addition to x, it also provides Z as the input Gauss noise generator. In the first experiment, we did not find an effective strategy - we just learned to ignore the noise generator - which is consistent with Mathieu. On the contrary, for our final model, we only provide noise from the form, which is applied to our generators in multi-layer training and testing time. Although the noise has declined, the network output we observed is only slightly random. The high random output conditions designed by GAN and the complete entropy distribution modeling to capture their conditions are an important problem in the current work.

B. Network architectures

We use the network to adjust the convolution depth generated for our generator and discriminator architecture, and use the convolution BatchNorm-ReLu module for the unsupervised learning of the generator and the system structure of the generator and the discriminator discriminator.

C. *Generator with skips*

One of the defining characteristics of image to image conversion problems is that they import high resolution into high resolution output grid grids. In addition, we consider the different appearance of input and output surfaces, but both are the same rendering of the underlying structure. As a result, the structure of the input and output is roughly aligned. We consider the design of the generator architecture around these.

There are many ways to solve the problem in the encoder decoder of the network. In this network, input is reversed through a series of gradually sampled layers until the bottleneck layer. The network requires all information to flow through all layers, including bottlenecks. For many images conversion problems, input and output share a large amount of low-level information and transmit information directly in the network. For example, in the case of color images, the input and output of edge positions are shared.

In order to provide a way to bypass the bottleneck to get this information to the generator, we follow the general shape of U-Net "add skipping links. Specifically, we add skip between each layer I and layer Ni connection, where n is the N-i layer for every channel and channel that simply skips a simple connection to the i layer.

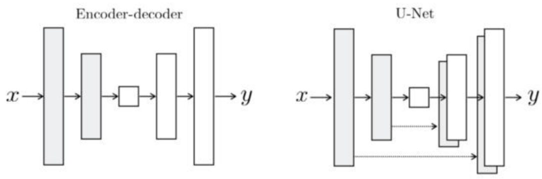

Figure 2: two options generator architecture. "U-Net" is the encoder decoder, skipping connections between the encoder and decoder in the mirror layer of the stack. [16]

D. Markovian discriminator (PatchGAN)

Although L2 losses and L1 losses are well known, their frequency resolution is not high, but in many ways they can still accurately capture frequency. For this problem, we do not need a new framework to enforce the correctness of low frequency.

The GAN discriminator only constrain the high-frequency structure simulation and relies on L1 to enforce the correct frequency. In order to simulate high frequency, we can limit our attention to the local structure of image blocks. Therefore, we have designed a PatchGAN discriminator Architecture - to punish only in the patch scale structure. For each N * N ,discriminator image classification, the N patch is classified as true or false. We run convolution discriminator in the image, and the average output is D to respond to all.

The effective image is modeled as a Markov random field, and the independence between pixels is assumed to be greater than the patch diameter. This relationship has been explored in real-time texture synthesis based on Mark off, and the common assumptions are texture and style models. Therefore, the loss of our PatchGAN form can be understood as texture / style.

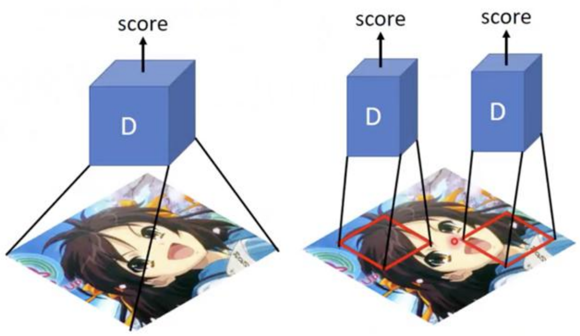

Figure 3: Shows old discriminator left detecting the image by the whole image and the Markovian discriminator detecting the image by the N by N part.

IV.EVALUATION AND RESULTS

In this research we recover the images based on CKDB (The Extended Cohn-Kanade Dataset) data set, first we need to degrade the image in different ways and different degrees, I combine the Motion Blur, Gaussian Blur, Bilateral Filter, Salt-and-Pepper Noise in different ways and different levels.

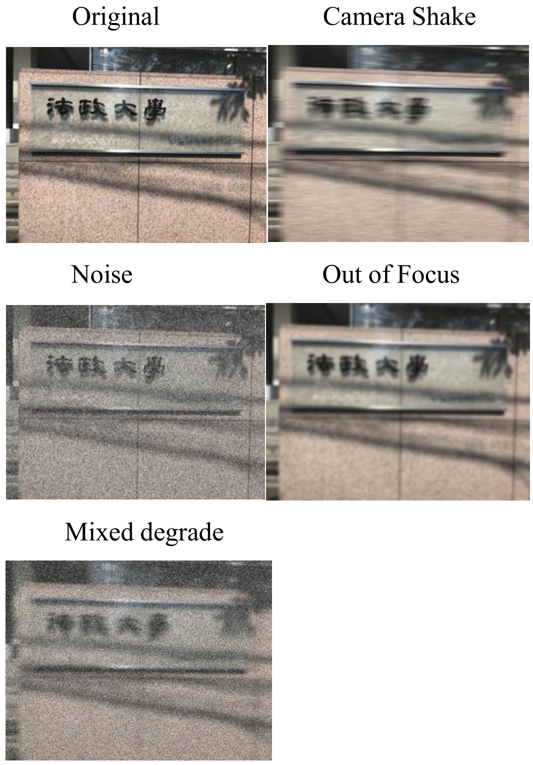

Figure 4: Shows the example of the original image and all the different types of degraded images when we take images in real life.

And then for training the model we need to resize the image into 256 pixels x 256 pixels and combine the original image and the degraded image side by side,and the clear image in the left side and the degraded image in the right side like Figure 5.

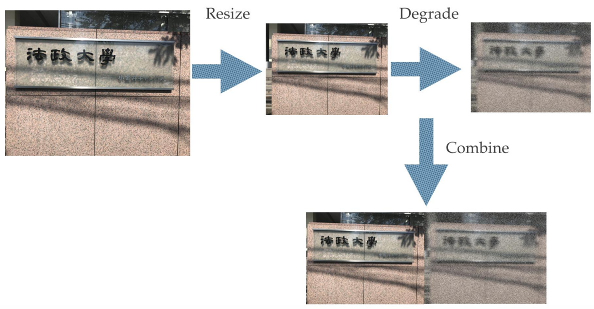

Figure 5. Shows how to recover the degraded images by using our cGAN based solution, resize the image first and combine the original and degraded image into a pair.

And after the prepare the pair data set, then split the combined data into train set and test set. And put them into the network to train the model. After the training, we got and trained model. The use the test set and the corresponding checkpoint to test the recovery result, in this thesis we compare the result with another generative adversarial networks DeblurGans to show that our method is good enough for the recover compositely degraded image because right now DeblurGans is one of the best networks for the blurred image, and we measure the quality of images by two parameters PRSN (Peak Signal to Noise Ratio) and SSIM (structural similarity).

For PRSN, the general value range is between 20 to 40. The larger the value, the better the image quality. For SSIM ,the general value range is from 0 to 1. The larger the value, the better the image quality. And Here is how to calculate PSNR and SSIM.

PSNR:

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (X(i,j) - Y(i,j))^2$$

$$PSNR = 10 \log_{10}\left(\frac{(2^n - 1)^2}{MSE}\right)$$

SSIM:

$$l(X,Y) = \frac{2\mu_X \mu_Y + C_1}{\mu_X^2 + \mu_Y^2 + C_1} \quad c(X,Y) = \frac{2\sigma_X \sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2}$$

$$s(X,Y) = \frac{\sigma_{XY} + C_3}{\sigma_X \sigma_Y + C_3}$$

$$SSIM(X,Y) = l(X,Y)\ c(X,Y)\ s(X,Y)$$

The main results of the experiment are down below:

1. Recovery the image from images with Motion Blur kernel size equals (15,15).

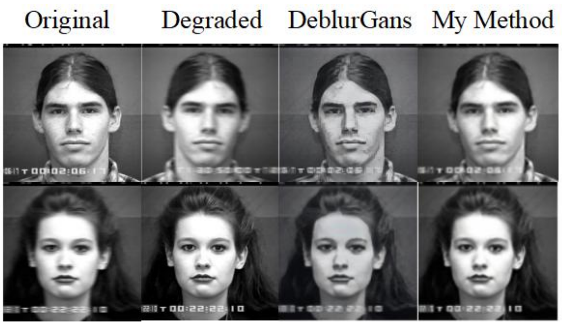

Figure 6. Shows the original image, the degraded image and the results of recover image by using our cGAN-based solution and DeblurGansin single motion blur with kernel size (15,15).

TABLE I. THE PSNR AND SSIM OF EXPERIMENT 1

	PSNR	SSIM
DeblurGans	39.5013	0.9579
My Method	39.7746	0.9886

From Table I we can see that the image is blurred by the light single motion blur which kernel size is (15,15) , the difference of PSNR is 0.273 and the difference of SSIM is 0.033, It is pretty close which means it is not a big difference when we recover the single light degraded image by the either our cGAN-based solution or DeblurGans.

2. Recover the image from images with Motion Blur kernel size equals (20,20).

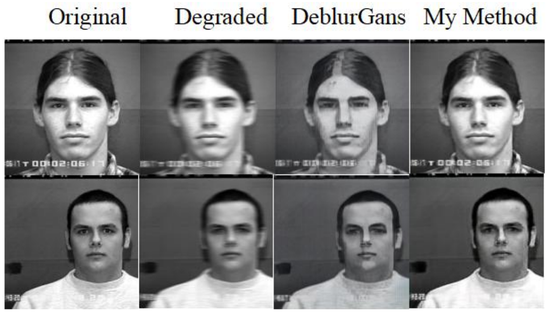

Figure 7. Shows the original image, the degraded image and the results of recover image by using our cGAN-based solution and DeblurGansin single motion blur with kernel size (20,20).

TABLE II. THE PSNR AND SSIM OF EXPERIMENT 2

	PSNR	SSIM
DeblurGans	38.5207	0.9576
My Method	39.5532	0.9772

From Table II we can see that the image is blurred by the light single motion blur which kernel size is (20,20), the difference of PSNR is 1.0325 and the difference of SSIM is 0.0196. In the experiment, the degrade is the same type of experiment 1 but with a bigger kernel size which means the image is more blurred. And the difference between our cGAN-based solution and DeblurGans start to become big. The our cGAN-based solution start showing its advantages.

3. Recover the image from images with Motion Blur kernel size equals (15,15) plus Gaussian Blur kernel size equals (15,15) and sigma equals 1.5 plus BilateralFilter.

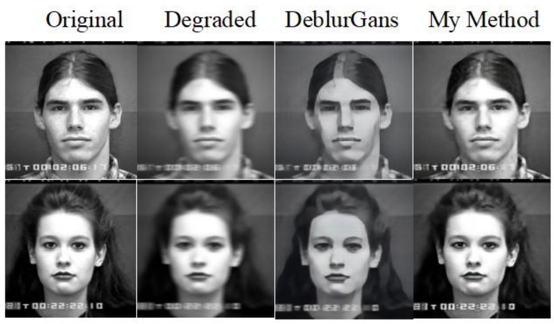

Figure 8. Shows the original image, the degraded image and the results of recover image by using our cGAN-based solution and DeblurGansin motion blur, gaussian blur plus bilateralfilter.

TABLE III. THE PSNR AND SSIM OF EXPERIMENT 3

	PSNR	SSIM
DeblurGans	38.3048	0.9509
My Method	39.5221	0.9778

From Table III the image is degraded by three different type blur the motion blur and gaussian blur and bilateralfilter, and the difference of PSNR is 1.2173 and the difference of SSIM is 0.0269. Based on the results, we can more clearly see that when the degrade is mix of three types blur, the difference of PSNR and SSIM is bigger then the results in experiment And it shows when the degraded is mixed more types, our method our cGAN-based solution is much better than DeblurGans.

4. Recover the image from images with Motion Blur kernel size equals (15,15) plus Gaussian Blur kernel size equals (15,15) and sigma equals 1.5 plus salt-and-pepper noise with SNR equals 0.02

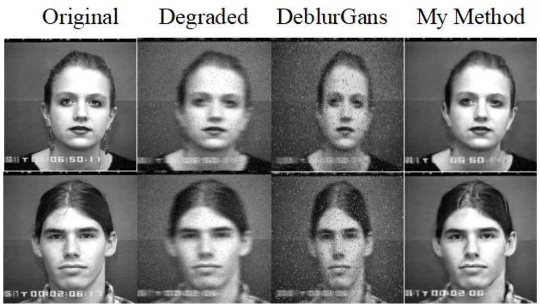

Figure 9. Shows the original image, the degraded image and the results of recover image by using our cGAN-based solution and DeblurGansin motion blur, gaussian blur, bilateralfilter plus salt-and-pepper noise.

TABLE IV. THE PSNR AND SSIM OF EXPERIMENT 4

	PSNR	SSIM
DeblurGans	37.5614	0.8312
My Method	39.4987	0.9782

From Table IV, image is degraded by two different type blur, the motion blur and gaussian blur and and salt-and-pepper noise.the difference of PSNR is 1.9373 and the difference of SSIM is 0.1470. Based on the results, we can see that when the image degraded by noise, our cGAN-based solution still can have good ability to recover the image to the original image as much as possible while the DeblurGans can not even remove the noise.

Based on the five experiments, by comparing the PSRN and SSIM of each experiment we can clearly see that when the degrade is mixed on a image, our method is much better than DeblurGans .

V. CONCLUSION

In this paper, we use the face database to compare the our cGAN-based solution method and DeblurGans method to recover the mixed degraded images problems.

We not only learn the mapping from the input image output image from the network, but also for training the mapping of the learning loss function. It can be applied to the same common way of the very different problems from the traditional to the loss formula.

From the experiment 1 and 2, that we found when the degraded image is single and light(motion blur and kernel size equals (15,15)), the different between our cGAN-based solution and DeblurGans not very obvious. But when the degarded is mixed (motion blur & gaussian blur & bilateralfilter) the different of the results of the prsn and ssim are more much, even when your degraded is plus noise like experiment 4 (salt-and-pepper noise), the our cGAN-based solution can recover it while the DeblurGanscan not remove the noise. So basically when the blur is light and single, there is no big difference between our cGAN-based solution and DblurrGANs. But when the degraded of image is mixed or more heavy,the results of recovery of image by using our cGAN-based solution is much better.

REFERENCES

[1] Orest Kupyn Volodymyr Budzan, Mykola Mykhailych1, DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks,2018

[2] Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Gen- erative adversarial nets. In NIPS, 2014.

[3] R. Szeliski. Computer Vision: Algorithms and Applications. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[4] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. ACM Trans. Graph., 25(3):787–794, July 2006.

[5] O.Whyte,J.Sivic,A.Zisserman,andJ.Ponce.Non-uniform deblurring for shaken images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010.

[6] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010.

[7] Ahmed, Mohamed N., et al. "A modified fuzzy cmeans algorithm for bias field estimation and segmentation of MRI data." Medical Imaging, IEEE Transactions on 21.3 (2002): 193-199.

[8] MacQueen, James. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14. 1967.

[9] D. Perrone and P. Favaro. Total variation blind deconvolution: The devil is in the details. In EEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[10] A. Gupta, N. Joshi, C. L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In Proceedings of the 11th European Conference on Computer Vision: Part I, ECCV'10, pages 171–184, Berlin, Heidelberg, 2010. Springer-Verlag.

[11] J. Sun, W. Cao, Z. Xu, and J. Ponce. Learning a Convolutional Neural Network for Non-uniform Motion Blur Removal. 2015.

[12] A. Chakrabarti. A neural approach to blind motion deblurring. In Lecture Notes in Computer Science (including sub- series Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016.

[13] D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. Van Den Hengel, and Q. Shi. From Motion Blur to Motion Flow: a Deep Learning Solution for Removing Heterogeneous Motion Blur. 2016.

[14] M. Noroozi, P. Chandramouli, and P. Favaro. Motion Deblurring in the Wild. 2017.

[15] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. A Jan. 2017.

[16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. 2018