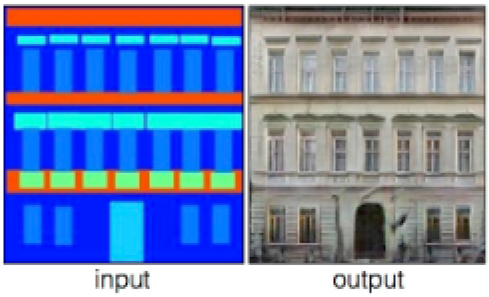
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, NAGPUR

PAPER TITLE: Image-to-Image Translation with Conditional Adversarial Networks

DEPARTMENT :	COMPUTER SCIENCE AND ENGINEERING
COURSE:	NEURAL NETWORKS AND DEEP LEARNING



TEAM MEMBERS:

BT19CSE044 RAGHAV AGRAWAL
BT19CSE047 NEHA KALBANDE
BT19CSE056 HEMANSHU CHAUDHARI

Abstract

The problem of image-to-image translation is studied and conditional adversarial networks(GAN) is used as a general-purpose solution to solve it. These networks usually learn a loss function so that mapping is trained ,which is from input image to output image. The same generic approach can be applied to problems that traditionally would require very different loss formulations. This approach can be effective at synthesizing photos from label maps, reconstructing objects from edge maps, and colorizing images, among other tasks. Due to many contributions from various online artists posting their own experiments with the system, further demonstrating its wide applicability and ease of adoption without the need for parameter tweaking. This research paper suggests that it can achieve reasonable results without hand-engineering the loss functions.

Method:

- GANs are generative models that learn a mapping from random noise vector z to output image y, G : z → y [24].
- Conditional GANs learn a mapping from observed image x and random noise vector z, to y, G : {x, z} → y.
- The generator G is trained to produce outputs that cannot be distinguished from “real” images by an adversarially trained discriminator, D, which is trained to do as well as possible at detecting the generator’s “fakes”.
- Designing conditional GANs that produce highly stochastic output, and thereby capture the full entropy of the conditional distributions they model, is an important question left open by the present work
- Both generator and discriminator use modules of the form convolution-BatchNorm-ReLu [29].

Objective:

The objective of a conditional GAN can be expressed as

$$LcGAN\ (G, D) = Ex,y[\log D(x, y)] + Ex,z[\log(1 - D(x, G(x, z))], \ (1)$$

where G tries to minimize this objective against an adversarial D that tries to maximize it, i.e.

$$G^* = \arg\ minG\ maxD\ LcGAN\ (G, D)$$

To test the importance of conditioning the discriminator, we also compare to an unconditional variant in which the discriminator does not observe x:

$$LGAN\ (G, D) = Ey[\log D(y)] + Ex,z[\log(1 - D(G(x, z))]. \ (2)$$

Previous approaches have found it beneficial to mix the GAN objective with a more traditional loss, such as L2 distance [43]. The discriminator's job remains unchanged, but the generator is tasked to not only fool the discriminator but also to be near the ground truth output in an L2 sense. We also explore this option, using L1 distance rather than L2 as L1 encourages less blurring:

$$LL1(G) = Ex,y,z[ky - G(x, z)k1]. (3)$$

Our final objective is

$$G^{*}= arg\ min\ G\ max\ D\ LcGAN\ (G, D) + \lambda LL1(G)$$

Builds on previous research
Past conditional GANs have acknowledged this by providing Gaussian noise z as an input to the generator, in addition to x (e.g., [55]). In initial experiments, this strategy wasn't found effective i.e., the generator simply learned to ignore the noise which is consistent with Mathieu et al [40].

Differs from previous work
Each session was tested just one algorithm at a time. Turkers weren't allowed to complete more than one session. Around 50 Turkers evaluated each algorithm. Unlike [62], we did not include vigilance trials.

Markovian discriminator (PatchGAN):

- It is well known that the L2 loss – and L1, see Figure 4 – produces blurry results on image generation problems [34]
- These losses fail to encourage high frequency crispness, in many cases they accurately capture the low frequencies.
- This is advantageous because a smaller PatchGAN has fewer parameters, runs faster, and can be applied to arbitrarily large images.
- Such a discriminator effectively models the image as a Markov random field, assuming independence between pixels separated by more than a patch diameter.
- Our PatchGAN can be understood as a form of texture/style loss

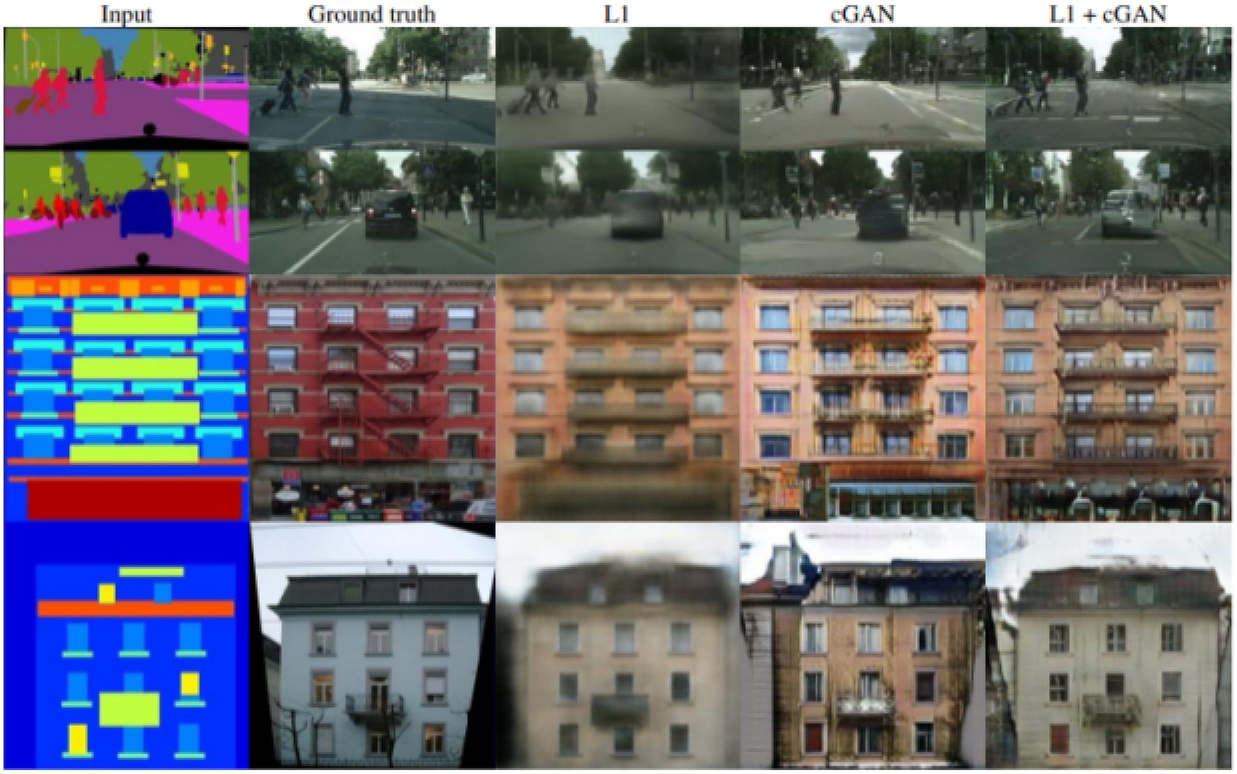


Figure 4: Different Losses induce different quality of result.

Optimization and inference:

- We follow the standard approach from [24]: we alternate between one gradient descent step on D, one step on G.
- We run the generator net in exactly the same manner as during the training phase
- This differs from the usual protocol in that we apply dropout at test time, and we apply batch normalization [29] using the statistics of the test batch, rather than aggregated statistics of the training batch.
- This approach to batch normalization, when the batch size is set to 1, has been termed “instance normalization” and has been demonstrated to be effective at image generation tasks [54].
- We use batch sizes between 1 and 10 depending on the experiment.

Evaluation metrics:

- Evaluating the quality of synthesized images is an open and difficult problem [52]. Traditional metrics such as perpixel mean-squared error do not assess joint statistics of the result, and do not measure the very structure that structured losses aim to capture.
- AMT perceptual studies For our AMT experiments, we followed the protocol from [62]: Turkers were presented with a series of trials that pitted a “real” image against a “fake” image generated by our algorithm.
- For map↔aerial photo, the real and fake images were not generated from the same input, in order to make the task more difficult and avoid floor-level results.

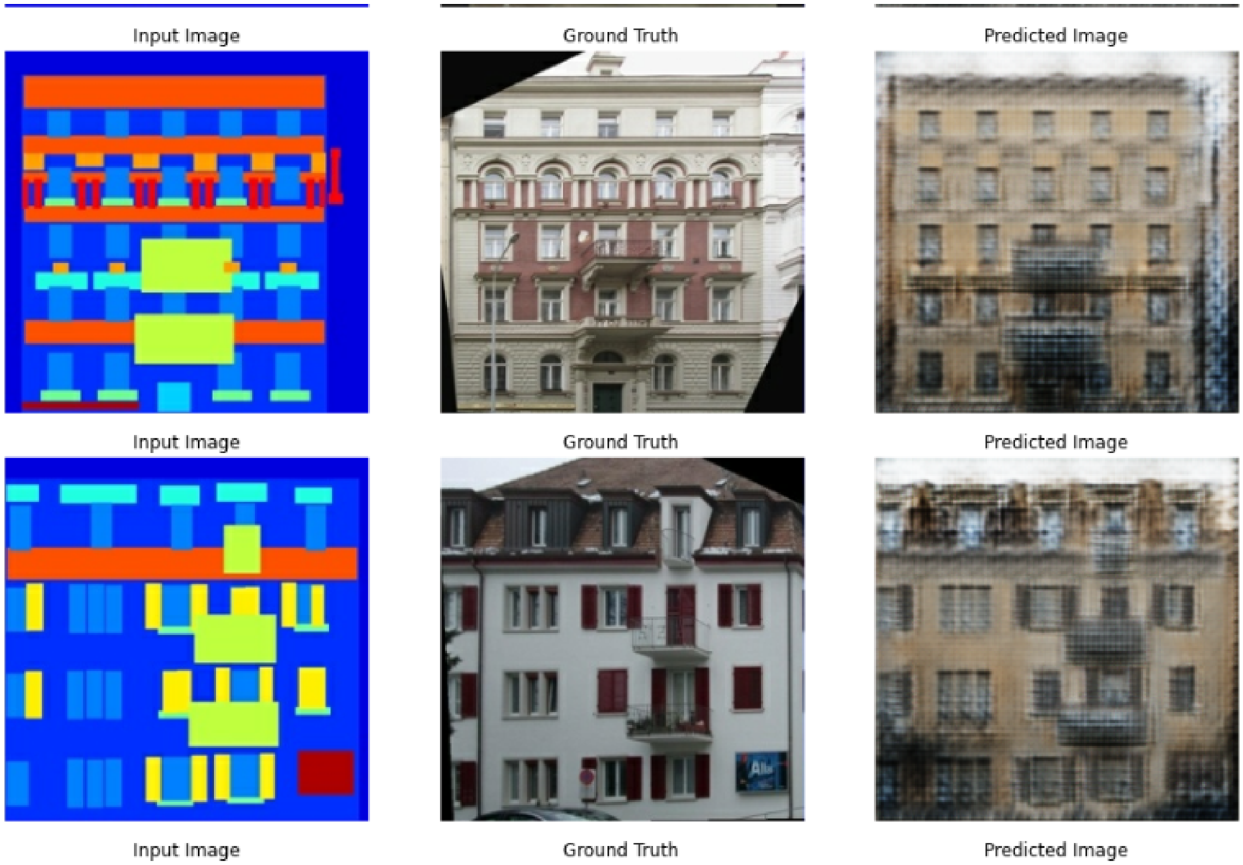
- We trained and tested on 256 × 256 resolution images and presented the results to Turkers at this same resolution.
- The intuition is that if the generated images are realistic, classifiers trained on real images will be able to classify the synthesized image correctly as well
- To this end, we adopt the popular FCN-8s [39] architecture for semantic segmentation, and train it on the cityscapes dataset.
- We score synthesized photos by the classification accuracy against the labels these photos were synthesized from

Builds on previous research

Weights were initialized from a Gaussian distribution by taking mean as 0 and standard deviation as 0.02. Cityscapes labels→photo 2975 training images from the Cityscapes training set [12], trained for 200 epochs, with random jitter and mirroring.

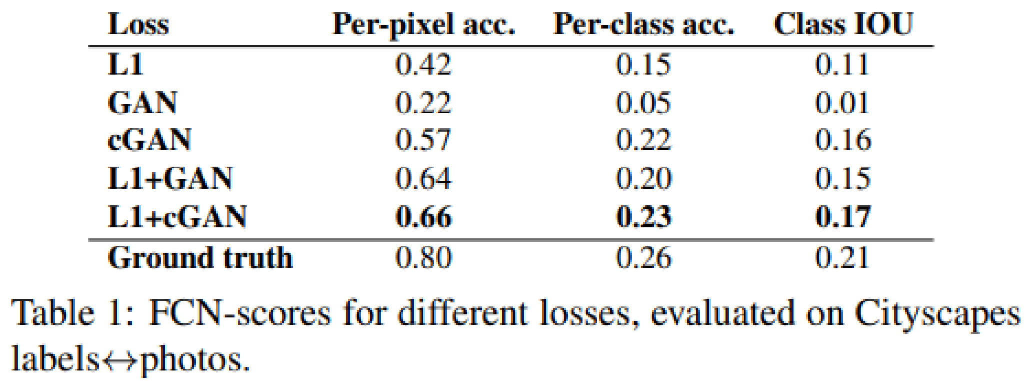
Differs from previous work

Data were split into train and test randomly. Day→night 17823 training images extracted from 91 webcams, from [33] trained for 17 epochs, batch size 4, with random jitter and mirroring.



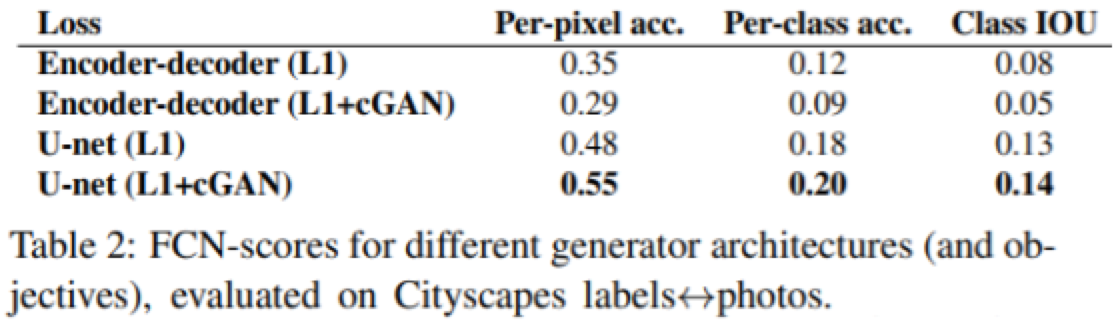
Loss:

Loss computed for different model articture using FCN scores



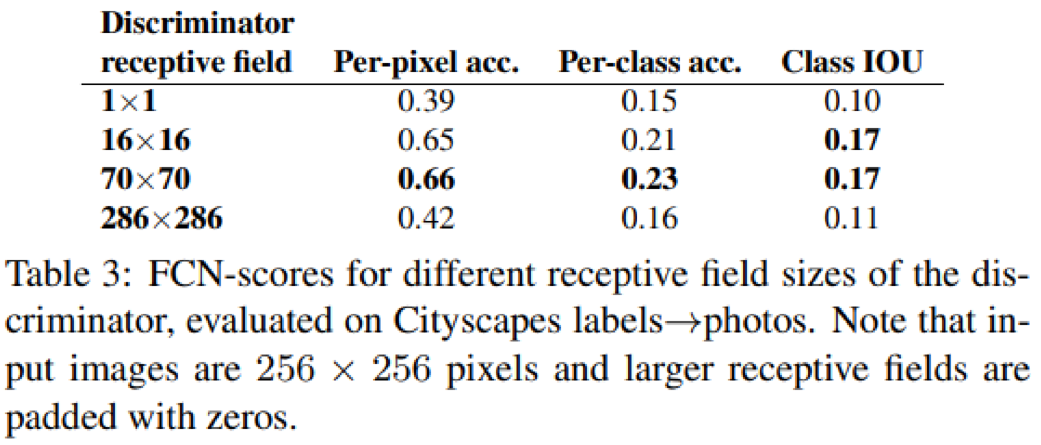
Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	0.42	0.15	0.11
GAN	0.22	0.05	0.01
cGAN	0.57	0.22	0.16
L1+GAN	0.64	0.20	0.15
L1+cGAN	**0.66**	**0.23**	**0.17**
Ground truth	0.80	0.26	0.21

Table 1: FCN-scores for different losses, evaluated on Cityscapes labels↔photos.



Loss	Per-pixel acc.	Per-class acc.	Class IOU
Encoder-decoder (L1)	0.35	0.12	0.08
Encoder-decoder (L1+cGAN)	0.29	0.09	0.05
U-net (L1)	0.48	0.18	0.13
U-net (L1+cGAN)	**0.55**	**0.20**	**0.14**

Table 2: FCN-scores for different generator architectures (and objectives), evaluated on Cityscapes labels↔photos.



Discriminator receptive field	Per-pixel acc.	Per-class acc.	Class IOU
1×1	0.39	0.15	0.10
16×16	0.65	0.21	**0.17**
70×70	**0.66**	**0.23**	**0.17**
286×286	0.42	0.16	0.11

Table 3: FCN-scores for different receptive field sizes of the discriminator, evaluated on Cityscapes labels→photos. Note that input images are 256 × 256 pixels and larger receptive fields are padded with zeros.

Obstacles faced by Conventional methods:

- There are Many problems in image processing, computer vision, computer graphics that can be posed as ‘translating’ an input image into a corresponding output image
- Structured losses for image modeling Image-to-image translation problems are often formulated as regression or per-pixel classification (e.g., [39, 58, 28, 35, 62])
- Generator with skips means defining feature of image-to-image translation problems is that they map a high resolution input grid to a high resolution output grid.

- For many image translation problems, there is a great deal of low-level information shared between the input and output, and it would be desirable to shuttle this information directly across the net.
- Amazon Mechanical Turk (AMT) perceptual studies For our AMT experiments, we followed the protocol from [62]: Turkers were presented with a series of trials that pitted a “real” image against a “fake” image generated by our algorithm.

Result:

The results of this research paper suggest that conditional adversarial networks (GAN) are a promising approach for many image-to-image translation tasks, especially for those involving highly structured graphical outputs. These networks have learned a loss adapted to the task and data at hand, which makes them applicable in a wide variety of settings.

Method used in this paper with L1+cGAN loss has fooled the participants on 22.5% of trials. conditional GAN is applied to semantic segmentation. The scores of conditional GAN were similar to the L2 variant of [62] (difference insignificant by bootstrap test), unfortunately fell short of [62]’s full method, which has fooled participants on 27.8% of trials in the experiment.

References:

[12] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In CVPR, 2016. 4, 16

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014. 2, 4, 6, 7

[28] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. ACM Transactions on Graphics (TOG), 35(4), 2016. 2

[29] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015. 3, 4

[33] P.-Y. Laffont, Z. Ren, X. Tao, C. Qian, and J. Hays. Transient attributes for high-level understanding and editing of outdoor scenes. ACM Transactions on Graphics (TOG), 33(4):149, 2014. 1, 4, 16

[34] A. B. L. Larsen, S. K. Sønderby, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In ICML, 2016. 3

[35] G. Larsson, M. Maire, and G. Shakhnarovich. Learning representations for automatic colorization. ECCV, 2016. 2, 8, 16

[39] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015. 1, 2, 5

[40] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. ICLR, 2016. 2, 3

[54] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022, 2016. 4

[55] X. Wang and A. Gupta. Generative image modeling using style and structure adversarial networks. In ECCV, 2016. 2, 3, 5

[58] S. Xie and Z. Tu. Holistically-nested edge detection. In ICCV, 2015. 1, 2, 4

[62] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. ECCV, 2016. 1, 2, 5, 7, 8, 16