Metrics on Image-to-Image Translation

Jianbo Chen(25001997)
Department of Statistics
University of California, Berkeley

Yuting Ye (26965918)
Divison of Biostatistics
University of California, Berkeley

Yaoyang Zhang (3032114788)
Department of Civil and Environmental Engineering
University of California, Berkeley

Abstract

We investigate appropriate metrics for the problem of image-to-image translation. We propose a metric for object transfer and a metric for style transfer. We evaluate our metrics on CycleGAN with various numbers of training epochs on different types of data sets and show the metric is highly correlated with human perception.

1 Introduction

The goal of image-to-image translation is to learn a mapping between an input image and an output image. The problem can be further divided into several types. The first type has no training data and only one image is given for the algorithm. The objective is often defined clearly. For example, classical image denoising [4] is within this class. The second type contains training data with paired images. There are many works along this line [3, 8, 11, 10, 14]. The conditional GAN was used for this type of task by Isola et al. [10]. Isola et al. [10] also proposed a metric called FCN-score that measures the quality of image-to-image translation. The third type contains training data with unpaired images. In the latter two types, the objective itself is usually not defined clearly and needs to be learned. This leads to the difficulty of evaluating a concrete method on such tasks, together with the difficulty of evaluating the fundamental difficulty of such a task itself. In this report, we aim to tackle such problems for the third type of image-to-image translation.

2 Related Work

2.1 Inception Score

Our methods were partly motivated by the idea behind Inception score [12]. Salimans et al. [12] introduced Inception Score as a metric to evaluate the performance of an image generator. It is empirically shown to be highly correlated with human judgement. The idea behind Inception Score is to apply a pre-trained state-of-the-art model on ImageNet [1], like Inception model [13] or ResNet [7] to every image generated from a given generator to get a conditional label distribution $p(y|x)$. Meaningful images should have a low entropy on the distribution $p(y|x)$. Moreover, as the variability of a generator is also concerned, $p(y) = \int p(y|x = G(z))dz$ should have a high entropy. Combining the two, the metric they propose is

$$\exp\{\mathbb{E}_x \mathrm{KL}(p(y|x)\|p(y))\}. \tag{1}$$

2.2 Cycle GAN

Cycle GAN [15] will be the main model where we evaluate our metrics. The advantage of Cycle GAN is learning a map between two domains instead of two specific images using adversarial approaches.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

The loss of a Cycle GAN is a adversarial loss plus a cycle consistency loss. Suppose F and G are maps between the two domains parameterized by neural networks. For each domain, the GAN loss is put on both real images and transferred images. For example, domain Y has the following GAN loss:

$$L_{\text{GAN}Y} = \mathbb{E}_{y \sim p_{\text{data}(y)}}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}(x)}}[\log(1 - D_Y(G(x)))]. \tag{2}$$

The cycle consistency loss L_{cycle} is the expectation of l_1 loss between $F(G(x))$ and x plus the l_1 loss between $G(F(y))$ and y where x, y are from the two datasets respectively. Then the generator aims to minimize:

$$L_{\text{GAN}X} + L_{\text{GAN}Y} + \lambda L_{\text{cycle}}. \tag{3}$$

2.3 Previous metrics

For specific tasks like image denoising or tasks with paired datasets, there are underlying truths (the target images). So one can evaluate the performance of a model easily by l_2 loss of raw pixels between the generated and the target image, more sophisticated similarity metrics like PSNR and SSIM [9], or the l_2 loss of extracted features of the two images. Isola et al. [10] also proposed a metric called FCN-score that measures the quality of image-to-image translation. None of the above metrics are suitable for methods targeted at unpaired data sets.

3 Methods

The following definition formally depicts the problem of image-to-image translation.

Definition 1. *Given two datasets A and B of images, under the assumption that the images in A and B lie in submanifolds $\mathcal{X}$ and $\mathcal{Y}$ respectively, one is asked to find a map $F : \mathcal{X} \to \mathcal{Y}$ so that for any $x \in \mathcal{X}$, $F(x) \in \mathcal{Y}$ and certain properties of x is preserved.*

The transferability between A and B, or more appropriately, $\mathcal{X}$ and $\mathcal{Y}$ depends on which properties of $\mathcal{X}$ should be invariant under the underlying F, and which properties should be transferred. Theoretically characterizing the difficulty is beyond our ability, so we seek an empirical approach of estimating the transferability.

Each layer of a pre-trained convolutional neural network can be taken as a feature bank. It was empirically shown by Gatys et al. [5] that across layers the texture representations increasingly capture the statistical properties of natural images while making object information more and more explicit. Lower levels in the network tend to capture the texture features from an image while higher levels tend to capture semantic information about an image. Given two datasets A and B, we can calculate the histogram of features on each layer of the network for each dataset.

3.1 Object transfer and style transfer

We define two classes of problems in image-to-image transfer: object transfer and style transfer. Object transfer aims to find a map $F : \mathcal{X} \to \mathcal{Y}$ that only transfers objects within a certain category to objects within another category but keeps other properties of an image, like objects in other categories, the locations of objects invariant. Style transfer aims to render an object with a texture taken from a different object [2], but keeps other properties like shape, category and the location of objects invariant.

Intuitively, suppose A and B have a different distribution of higher-layer features but a similar distribution of lower-layer features. Then the difference between the dataset A and the dataset B mainly lies in contents. We call this class of problem "object transfer". Concrete examples include the translation between apples and oranges, horses and zebras and dogs and cats that were provided by Zhu et al. [15].

Otherwise, suppose A and B have a different distribution of lower-layer features but a similar distribution of higher-layer features. Then the difference between the dataset A and the dataset B mainly lies in textures. We call this class of problem "style transfer". Concrete examples include the translation between paintings and photos, summer and winter Yosemite photos provided by Zhu et al. [15].

3.2 Evaluation of methods

In this section we propose metrics for object transfer and style transfer respectively. Below we describe the metric on a single image under transfer in detail. The overall metric is got by averaging over all images in a dataset.

3.2.1 Object-transfer metric

Assume A and B contains objects in category i and j respectively with i, j lying among the $1,000$ classes of the ImageNet [1]. Our goal is to transfer an image x in the submanifold $\mathcal{X}$ where A lies to an output image in the submanifold $\mathcal{Y}$ where B lies, so that objects in i should be transferred to objects in j but other properties of x should be kept invariant. For an ideal map F, the conditional distribution over labels given an image should have the following property:

$$\begin{aligned} p(y = i|F(x)) &= p(y = j|x); \\ p(y = j|F(x)) &= p(y = i|x); \\ p(y = k|F(x)) &= p(y = k|x) \text{ for } k \neq i, j. \end{aligned}$$

Naturally, if we define $\tilde{p}(y|x)$ as a conditional distribution that swaps the value on i and j but keeps other categories the same as $p(y|x)$. We propose to use $D_{KL}(\tilde{p}(y|x)||p(y|F(x)))$ as a metric, where D_{KL} is the KL divergence between distributions. We call this metric the swapping KL divergence.

3.2.2 A metric for style transfer

Assume the task is style transfer between A lying on $\mathcal{X}$ and B lying on $\mathcal{Y}$. Suppose we apply the tranfer map F to an image $x \in A$. We wish the texture of $F(x)$ is similar to the texture of images in B but the other properties including the categories of objects are invariant. So our metric should encourage texture similarity between $F(x)$ and images in B but penalize object inconsistency between x and $F(x)$. So our metric can be defined as a combination of the following two:

$$M_1 = \frac{d(p(F_l|x), p(F_l|B)) - \sum_{b \in B} d(p(F_l|b), p(F_l|B))/|B|}{\text{Var}\{d(p(F_l|x), p(F_l|B))\}}, \tag{4}$$

where F_l are features at layer l , $p(F_l|B)$ is defined as the histogram of F_l evaluated on the entire dataset B and d is the χ^2-distance between distributions. The variance is the sample variane of χ^2 distance between the histogram of each single image in B with the histogram of the entire dataset B. (This reminds one of the t-test in statistics.) This metric characterizes similarity of texture between x and B, and

$$M_2 = d(p(y|x), p(y|F(x))). \tag{5}$$

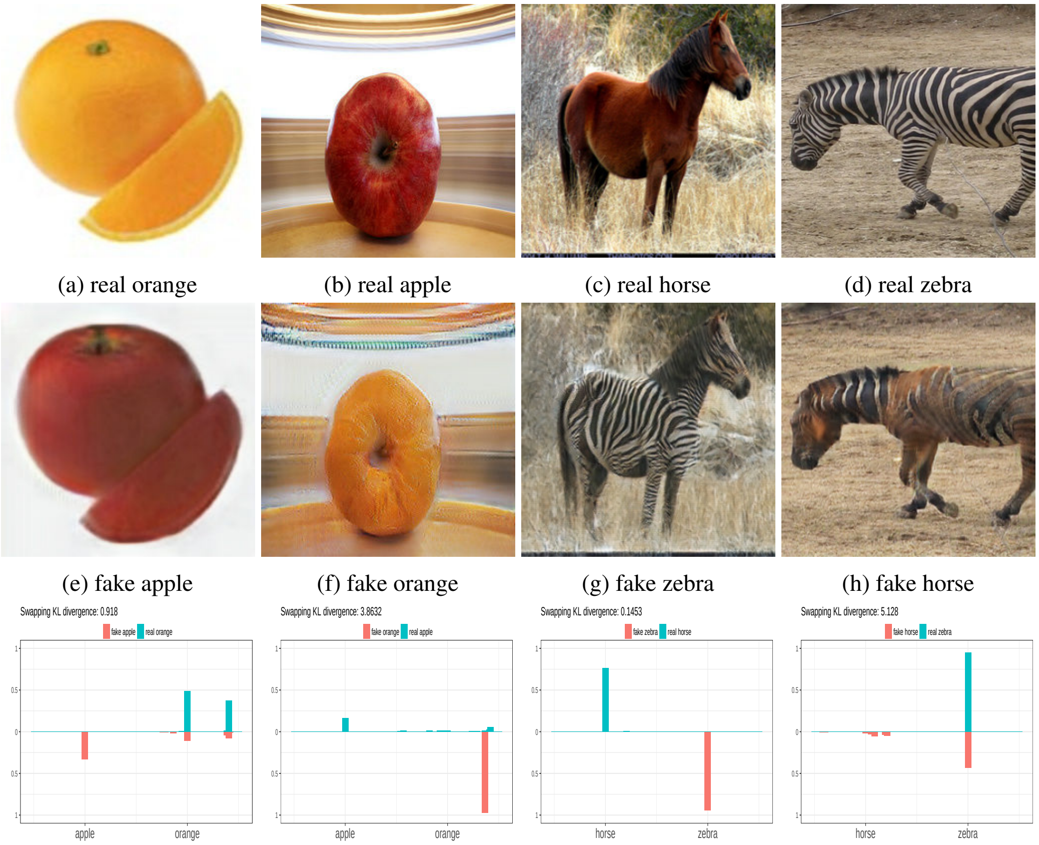
is the similarity between the label distribution of the original pictures and the transferred pictures. A good style transfer should not change the label distribution so a small M_2 is desired.

However, unlike object transfer, a single M_1 score for a single image sometimes is not persuasive and reliable enough to evaluate the performance of style transfer between two datasets. Thus, we propose a test to evaluate the similarity between the distribution of M_1 scores of fake images and the distribution of M_1 scores of real images. Our null hypothesis is that the mean value of the two distributions of M_1 scores are the same. If the p-value of this test is large, it means that our hypothesis is accepted and the the styles of the real and fake images are close enough. On the contrary, if the p-value is small, it means our model is bad.

4 Results

4.1 Object Transfer

We apply the swapping KL divergence to two examples from CycleGAN, (1) apple and orange, (2) horse and zebra, which are the representative results for the unpaired image-to-image translation method on the object-transfer task. Since ImageNet can misclassifies the horse as other looking-like animals, e.g., hartebeest, we extend the definition of "horse" in our setting to include sorrel horse,



(a) real orange (b) real apple (c) real horse (d) real zebra

(e) fake apple (f) fake orange (g) fake zebra (h) fake horse

(i) orange v.s. apple (j) apple v.s bagel (k) horse v.s. zebra (l) zebra v.s. zebra

Figure 1: Results of the object transfer task by CycleGAN on a single object. Top: real objects; middle: transferred objects; bottom: probability distributions of classification by ImageNet. Each subcaption of a figure in the bottom row shows the maximum likelihood object (the real v.s. the fake) indicated by ImageNet.

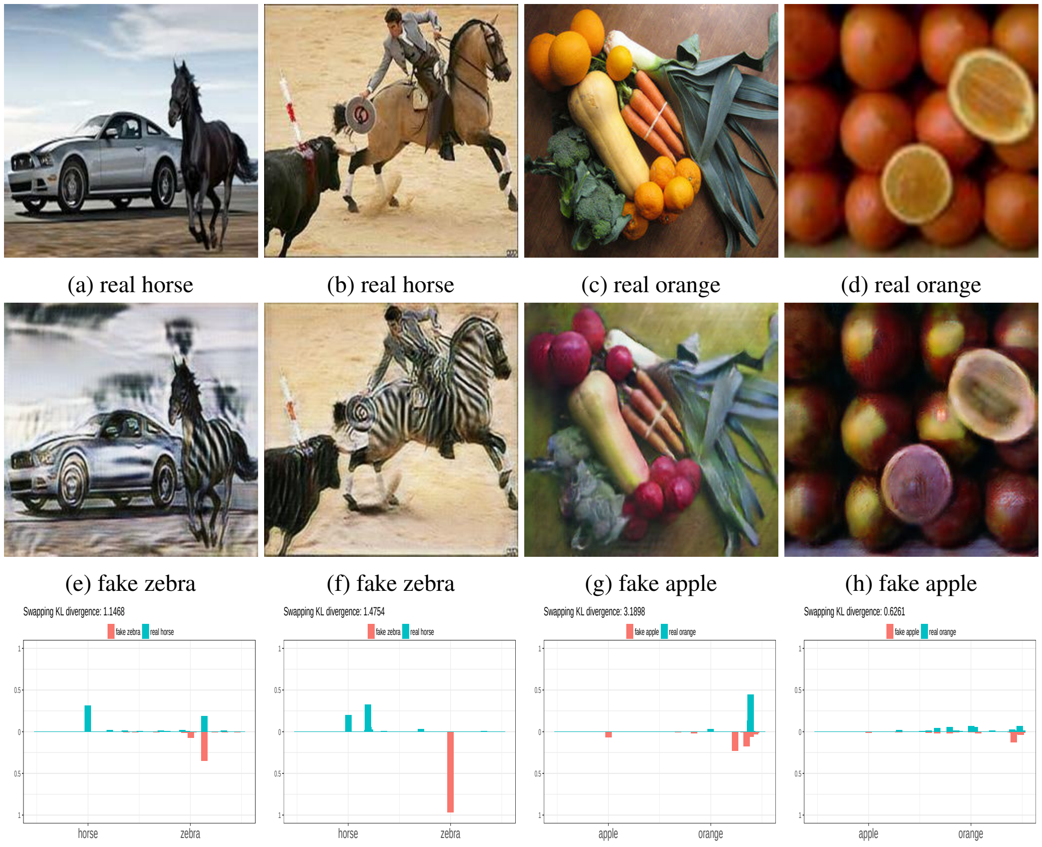
gazelle and hartebeest. Similarly, let “apple” include Granny Smith, strawberry, rose hip and let “orange” include orange and lemon. From our experience, the tasks with the swapping KL divergence less than 1, around 1, and highly larger than 1 can be evaluated as “good”, “ok” and “bad”.

4.1.1 Single Object

Figure 1 displays four selected figures with only one object. The first column is the translation from an orange to an apple, with the swapping KL divergence of 0.918. CycleGAN successfully translated the entire orange to an apple, but failed to translate the small piece. Both the real figure and the fake one have non-ignorable probability on the lemon class. In the second column, CycleGAN translated the apple to a bagel instead of an orange, and the corresponding swapping KL divergence is 3.8632. In the third column, the translation is very successful, with a small swapping KL divergence of only 0.1453. The distribution plot shows a clear swapping on the zebra class and the horse class. In the fourth column, the translation from the zebra to a horse obviously failed with a high swapping KL divergence of 5.128. These results show that our proposed metric for the object-transfer task, the swapping KL divergence, works well for a single object. It can accurately captures the key information of this specific task, i.e., translation of a single object.

4.1.2 Multiple Objects

In most occasions, we have multiple objects in one figure, but only want to translate one object while keeping others unchanged. The swapping KL divergence is motivated by such ideal case, and we



(a) real horse (b) real horse (c) real orange (d) real orange

(e) fake zebra (f) fake zebra (g) fake apple (h) fake apple

(i) horse and car v.s. car and zebra (j) ox and horse v.s. zebra (k) squash v.s.tray (l) corn and lemon v.s. hay and coral fungus

Figure 2: Results of the object transfer task by CycleGAN on multiple objects. Top: real objects; middle: transferred objects; bottom: probability distributions of classification by ImageNet. Each subcaption of a figure in the bottom row shows the top maximum likelihood objects (the real v.s. the fake) indicated by ImageNet.

select four figures to show the performance as shown in Figure 2. In the first column, the real figure contains a horse and a sports car. CycleGAN only targets at translating the horse into a zebra, and the resulting figure looks okay with the corresponding swapping KL divergence of 1.1468. From the distribution plot, it can be seen that the mass on the horse is shifted to the zebra and the sports car. The second column aims to translate the horse to a zebra while keeping the ox and the person untouched. The resulting figure looks okay, but ImageNet gives all the mass to the zebra and leaves nothing to the ox. Thus the swapping KL divergence is as high as 1.4754. In the third columns, there are numerous objects including orange, squash, carrot, and the target is only the orange. However, ImageNet only assigns limited mass to the orange and the apple on the real figure and the fake figure respectively. The resulting figure looks not bad but the swapping KL divergence is boosted to 3.1898. The fourth column is a translation from a bunch of oranges to a bunch of apples. The resulting figure is awful, but ImageNet can hardly distinguish these objects on both the real figure and the fake figure. The masses distribute equally on both figures, and thus the swapping KL divergence is unreasonably as low as 0.6261. From these results, we can see that our proposed metric does not work consistently under the scenario of multiple objects. The key reason lies in that ImageNet either classifies a bunch of objects as a complex, such as grocery and dishes on the tray instead of assigning a mass to each object; or assigns the majority of mass to a dominating object like the zebra.

4.1.3 Overall Performance

Since there is no ground truth about the generating figures, the loss of GAN does not convey information about the model performance as the loss of ImageNet for the classification task. The swapping KL divergence, on the other hand, can be also used to evaluate the model, in addition to evaluating a pair of real figure and fake figure. We have three models on the orange-apple task and the horse-zebra task: one trained over $1,000$ epochs labeled as "bad", one trained over $15,000$ epochs labeled as "ok" and one pre-trained model used by [15], labeled as 'good'. We apply the three models to the testing dataset, and compute the average of the swapping KL divergences. In Figure 3, the averaged metric decreases as the model gets better, which indicates that it can be used as an indicator of the model performance. Furthermore, the performance of CycleGAN highly relies on the specific objects for translation. For example, horse-to-zebra and apple-to-orange are easier than zebra-to-horse and orange-to-apple, since adding the strip or flesh texture is more straightforward than removing them. Figure 3 validates this argument.

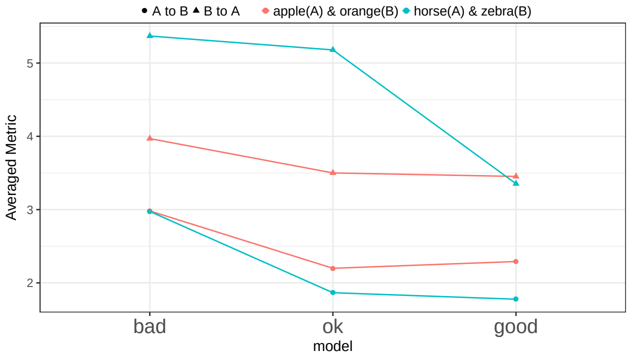


Figure 3: The averaged metric for three CycleGAN models on the testing dataset.

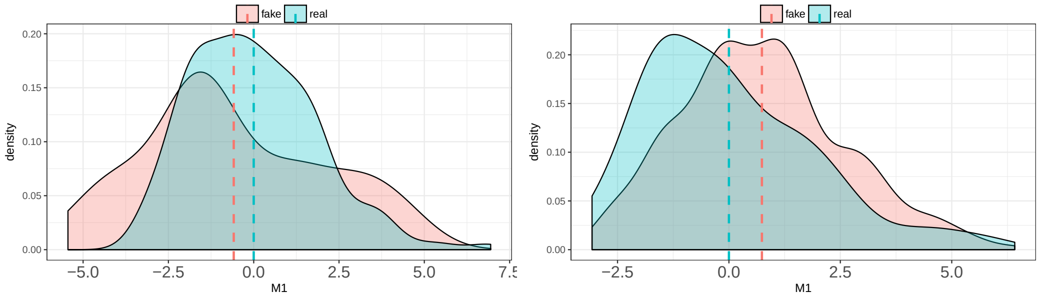
4.2 Style Transfer

As mentioned in Section 2, an ideal style transfer should keep object categories, shapes, positions and other high-level features invariant while making the low-level features, that is, its style, similar to those in the target datasets. We thus use a combination of M_1 and metric M_2 defined in Section 2.3.2 to evaluate the performance of the model. We assume that a good model would have a large M_1 value and a small M_2 value.

In this report, we only explore the performance of M_1 metric on style transfer problems. We apply M_1 metric to another example pair from CycleGAN, photos and Monet paintings. To get low-level features, we feed the sample images into an Inception v3 Net pre-trained on ImageNet and extract the intermediate results after each max-pooling layer for the first 5 convolutional layers. We compute the histogram of features using the idea from Histogram of Oriented Gradients (HOG) for each layer, and we end up only using the features from the second to the fifth layer because the features in the first layers are too basic to make a difference in different styles.

Figure 4 (a) shows the histogram of M_1 scores of real Monet paintings and fake Monet paintings generated from photos by CycleGAN. These two distributions have a p-value of 0.65, meaning that the means of the two distributions are not significantly different. Figure 5 shows a sample generated painting and a sample real painting. It is hard to tell which painting is fake if we are not a big fan of Monet, which agrees with our M_1 metric results.

Figure 4 (b) shows the histogram of M_1 scores of real photos and fake photos generated from Monet paintings by CycleGAN. The M_1 score (p-value) of these two distributions are 0.0004, meaning that their means are significantly different. Figure 5 shows a sample real photo and a sample generated photo. We can easily tell which is real and which is fake and again, our M_1 metric successfully

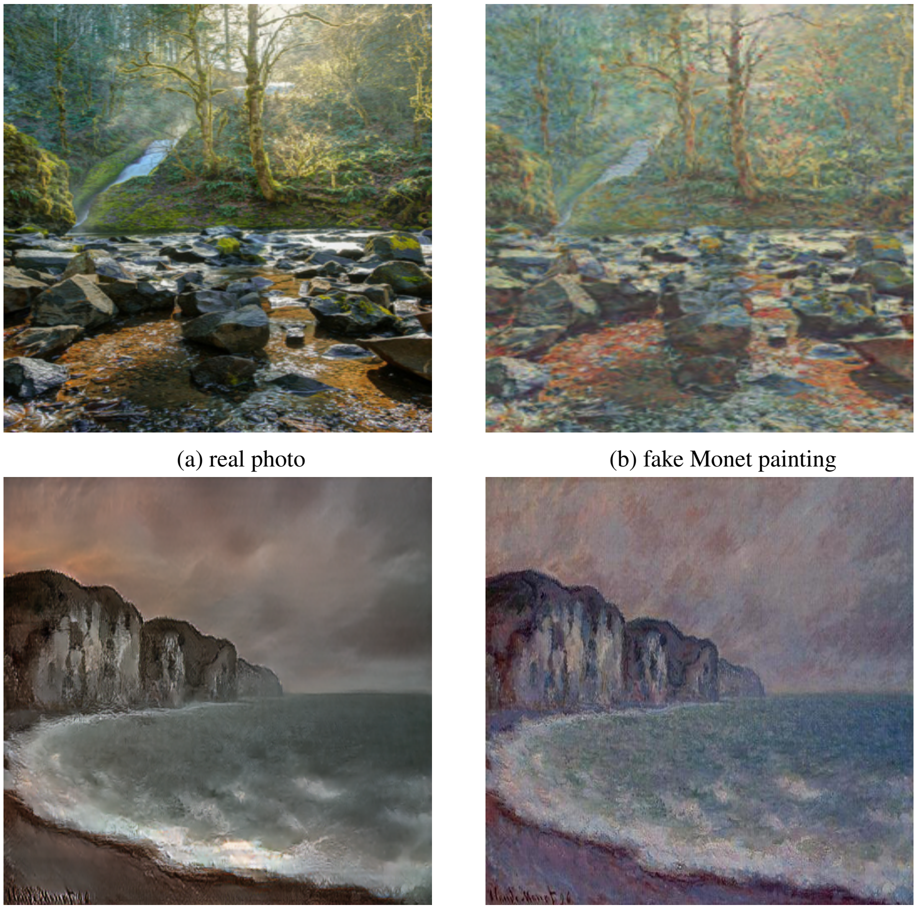


(a) Monet paintings

(b) photos

Figure 4: Distribution of χ^2 distances

captures this.



(a) real photo

(b) fake Monet painting

(c) fake photo

(d) real Monet painting

Figure 5: Results of the style transfer task by CycleGAN on photos and Monet painting. Top: A real photo and the Monet painting generated from that photo; bottom: fake photo generated from a Monet painting and the corresponding Monet paiting

Our metric can be used to test whether two datasets have the same style and thus is a good indicator of the performance of a style transfer model. For the specific photo and painting translation task, one can observe that generating paintings from photos is much easier than its reverse. This is because real photos often contains too much detailed information that is too subtle to be captured by the translating network.

5 Discussion and Future work

In this project, we proposed two metrics to evaluate the similarity between two figures or two imaging datasets in terms of the specific tasks. The swapping KL divergence is used for the object-transfer task and the combination of the M_1, M_2 metrics for the style-transfer task. We applied the two metrics to CycleGAN. For the object-transfer task, the swapping KL divergence performs well on a single object, but suffers from classification issues of ImageNet on multiple objects. Specifically, ImageNet either classifies a bunch of objects as a complex, e.g., grocery, or assigns a large proportion of masses to the most distinguishable object. To overcome this problem, one can split the entire task into multiple separate tasks: (1) detect the object regions, (2) compute the swapping KL divergence of each region, (3) take average of the divergences (illustrated by Figure 6). For example, one can utilize R-CNN [6] for this purpose. The regions are supposed to be proposed based on the real figure and directly used on the fake figure, in order to make the new metric sensitive to the location changes. As to the style-transfer task, the choice of the lager l needs careful selecting to give a good representation of the texture. Furthermore, we only focus on the M_1 metric in this report and skip the results for the combination of M_1 and M_2. The balance between the two metrics also requires substantial care.

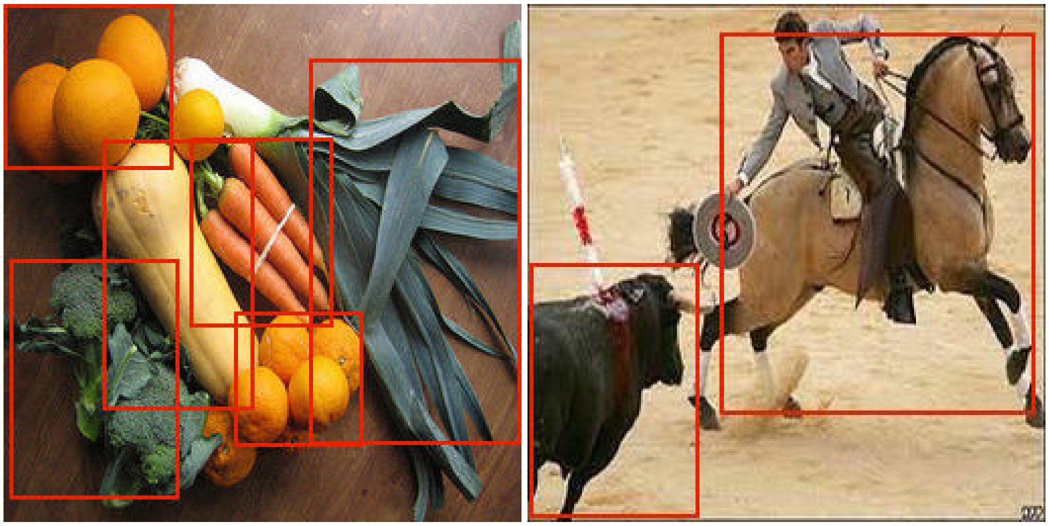


Figure 6: Region-based metric.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[2] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM, 2001.

[3] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2650–2658, 2015.

[4] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006.

[5] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 262–270, 2015.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):142–158, 2016.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[8] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340. ACM, 2001.

[9] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2366–2369. IEEE, 2010.

[10] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

[11] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[12] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2226–2234, 2016.

[13] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[14] Xiaolong Wang and Abhinav Gupta. Generative image modeling using style and structure adversarial networks. In *European Conference on Computer Vision*, pages 318–335. Springer, 2016.

[15] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.