10.1148/ryai.2021190181

Appendix E1

Quantitative Assessment Metrics

There are two main factors we wish to assess to estimate the quality of outputs from a trained generator network. One is the shape of the distributions, or how probable are the synthesized images were under the true data distribution, and the other is how large is the support of the generated distribution. Neither of these factors are straightforward to quantitatively assess and have been a subject of research since the advent of GANs.

The difficulty in assessing the fidelity to which the generated distribution follows the true data distribution stems from the fact that:

Images Are Unpaired

We wish to compare sets, as opposed to pairs of images for which most conventional image similarity metrics are designed.

Conceptual Assessment

The comparison is based on conceptual attributes of appearance that are inherently subjective and difficult to quantify.

The most popular quantitative methods as described in the literature are the following:

Inception Score (IS)

The first quantitative metric proposed in the literature. It typically involves presenting the images X to a trained Inception network (13) (usually ImageNet (14) pretrained).

Intuitively, high fidelity to the true distribution implies low entropy with respect to the conditional prediction p (y|**X**) (samples are unambiguous) and high distributional support translates to high entropy with respect to the marginal p (y) (samples have high diversity).

Most prominent criticisms of the metric focus on the fact that it is difficult or impossible to apply to a domain for which no pretrained network is available and that it does not explicitly utilize the real images in the computation of the score.

Frechet Inception Score (FIS)

An alternative to the IS proposed to ameliorate some of the aforementioned issues is the Frechet Inception Distance (FID). Both real and synthetic images are encoded into a discriminative feature space by means of a pretrained neural network. The overlap of those features is then assessed using the Frechet distance.

As a result the metric can be more robust to transferring to a slightly different domain than the one of the encoder network, as long as the features are still discriminative enough for the new domain.

Multiscale Structural Similarity Index (MS-SSIM)

The MS-SSIM is a metric that does not require the use of a pretrained model. It is based on the SSIM, which was designed to improve upon traditional image quality metrics (15,16) and has also been used as a loss function in deep learning as it is differentiable (17).

As SSIM is a distance metric between images, it is necessary to randomly pair the real and synthetic images (as no better pairing is available), compute the SSIM of each set and then compare with within set self-similarities.

Multiscale Sliced Wasserstein's Metric

An interesting alternative to conventional GAN evaluation metrics is the Multiscale Sliced Wasserstein metric (2). It is based on the concept of the sliced Wasserstein distance between distributions.

The Wasserstein's distance, also referred to as the "earth mover distance" can be computed by simple subtraction of paired samples. However, the samples must be optimally paired and finding the optimal transport policy between multidimensional distributions is very computationally intensive. In contrast, the optimal transport between two one-dimensional distributions can be simply found by sorting their samples. Based on this observation the sliced Wasserstein's is an approximation of the Wasserstein's distance as a finite sum of one-dimensions projections to sufficiently many random directions, where it is much simpler to find the optimal coupling.

Appendix E2

A Primer on GANs

Pixel resolution of GANs

Early research into GANs did not focus on increasing generated image resolution due to challenges with respect to training instability and resulting quality. For every increase in pixel size, there is a concordant increase in dimensionality and complexity, which makes the problem more difficult.

In addition, and despite the abundance of high-resolution medical image datasets, that is not necessarily the case for natural images, on which GAN literature relies. In fact, the authors in (2) resorted to synthetically increasing the resolution of the popular CelebA facial image dataset to train a GAN at pixel resolution as high as 1024 × 1024. To date there has been little published research focusing on high-resolution GANs in the medical imaging domain.

Progressive Growing of GANs

Despite the original concept of progressively growing of GANs presented in (2), this work included several further important contributions, including a) a dynamic weight initialization method proposed to equalize the learning rate between parameters at different depths, b) batch normalization was substituted with a variant of local response normalization to constrain signal magnitudes in the generator c) addition of the mini-batch standard deviation as an extra signal to the discriminator to promote intrabatch variability d) a new evaluation metric was proposed

(Sliced Wasserstein distance). Finally, further stability and performance gains stem from using the Wasserstein objective with gradient penalty (18).

Nevertheless, we experienced stability issues and mode collapse despite using progressive training, which we alleviated by taking the following measures: *(a)* adding supervised information (19) we conditioned on useful attributes, such as the view (craniocaudal or mediolateral oblique) and breast mask size, *(b)* decreasing the default learning rate (from 0.002 to 0.0015), *(c)* gradually increasing the discriminator iterations per each generator update, from 1 to a maximum of 5, *(d)* increasing the capacity of the networks, by doubling the number of neurons in the final feature layer and the starting depth of the network. We trained for 33 epochs, before resuming and presenting an additional 5 million images (sampled with replacement). We selected the best network checkpoint based on the sliced Wasserstein distance. The whole process took approximately 52 hours on an NVIDIA DGX-1, with 8 V100 GPUs (Fig E1).

Mode Collapse

Mode collapse is an important problem encountered in GAN research, difficult to detect or avoid. In a GAN framework the generator network is aiming to create a synthetic image x = G (z), where G is the generator, and z represents a feature vector drawn from a uniform distribution in the latent space. The resulting image x can therefore manifest as one of many variations of the underlying latent space. However, let us suppose a situation where the generator network is trained without sufficient updates to the discriminator. In this situation generated images may converge to an optimal image x*, independent of z; that is to say that the generated image will not be related to the underlying latent features:

$$x^* = argmax_x D(x).$$

Here, the 'mode' is said to have collapsed to a single point, as the gradient associated with z approaches zero. All the discriminator has to do to be successful is to find one (or few) x*, which is trivial, as that is all the generator is producing at this point. Continued training essentially overfits the generator to carry on producing x*, and the discriminator to finding x*, and the model cannot converge to equilibrium.

Beyond the most extreme manifestations of mode collapse, the problem is typically much more subtle and therefore difficult to detect. More generally, partial collapse would preclude the generator from producing images in some areas of the support of the real image distribution. Therefore, there would be areas of the real (target) distribution to which the synthetic (source) distribution would assign zero probability density.

Appendix E3

Mutual Information

The method for calculating mutual information is proposed in (18) and can be outlined as follows:

For each data point y_i in class c_i, we first find the distance to its' k^{th}-nearest neighbor (k = 3) within the same class and denote it as d_i. We then count the neighbors that lie within the

sphere with radius d_i regardless of their class and denote that count as m_i. The mutual information score for each data point i is given by the following equation (18):

$$I_i = \psi(N) - \psi\left(N_{c_i}\right) + \psi(k) - \psi(m_i),$$

where N is total number of data points, N_{c_i} is the number of data points belonging to class c_i, k is the number of neighbors we consider (default is 3) and m_i is the number of neighbors within a sphere of radius d_i.

To calculate the mutual information between the two groups of points, we need to pair their samples. To do so, we used the optimal coupling for one-dimensional distributions, which according to Optimal Transport theory is simply obtained by the sorting operator.

Appendix E4

Application Design

Our iOS application was developed for Apple iPad Pro using Unity 2018.2.2f1, built in Xcode 9.2 and deployed over Amazon S3. The iPad had a screen resolution of 2048 × 2732 pixels, a screen height of 12.9" and an aspect ratio of 4:3. The architecture was a basic model-view-controller (MVC) implementation binding a data-store (Model) to a simple state-machine (Controller) that manages the presentation and input from several pages (Views) presented to the player. The basic modes of the app in the controller layer, were: *(a)* survey, *(b)* tutorial, and *(c)* gameplay states.

During an onboarding survey participants were asked a variety of questions for the purpose of stratifying the results of the study. The tutorial instructed participants on how the real and synthesized mammograms would be presented and how to interact with the application. The gameplay state displayed the real and synthesized images to the participant and allowed them to select the "real" image.

In the view layer (Fig 4), the gameplay state of the app was designed to emulate a basic DICOM viewer displaying image pairs to the participant with a high level shading language (HLSL) shader to enable zooming and panning using familiar gestures. The participants were not able to change the window level. The images were selected at random from the pool, and assigned to the left and right of the screen on a 'coin-toss' upon presentation. Once presented an image was removed from the pool for that players session (sampled without replacement). The probability distribution for the coin-toss presentation of images was verified by a unit test which always chose the left image and asserted the mean average score to be 5 out of 10.

References

13. Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2015; 1–9 https://doi.org/10.1109/CVPR.2015.7298594.

14. Deng J, Dong W, Socher R, Li LJ, Li K, Li FF. ImageNet: a large-scale hierarchical image database. In: Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2009; 248–255.

15. Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier GANs. In: Precup D, Teh YW, eds. Proceedings of the 34th International Conference on Machine Learning. Vol 70. Sydney, Australia: ML Research Press, 2017; 2642–2651. https://dl.acm.org/doi/10.5555/3305890.3305954.

16. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 2004;13(4):600–612.

17. Godard C, Mac Aodha O, Brostow GJ. Unsupervised monocular depth estimation with left-right consistency. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2017; 270–279.

18. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC. Improved training of Wasserstein GANs. In: von Luxburg U, Guyon I, Bengio S, Wallach H, Fergus R, eds. Proceedings of the 31st International Conference on Neural Information Processing Systems. Cambridge, Mass: MIT Press, 2017; 5767–5777. https://dl.acm.org/doi/10.5555/3295222.3295327.

19. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X. Improved techniques for training GANs. In: Lee DD, von Luxburg U, Garnett R, Sugiyama M, Guyon I, eds. Proceedings of the 30th International Conference on Neural Information Processing Systems. Cambridge, Mass: MIT Press, 2016; 2234–2242. https://dl.acm.org/doi/10.5555/3157096.3157346.