Towards an Affine-Invariant Generative Model

Chris B. Dock [1] **Michael-Andrei Panaitescu-Liess** [1] **Ethan Remsberg** [1] **Niall Williams** [1]

Abstract

Generative adversarial networks (GANs) are neural networks that are commonly used in image synthesis tasks where realism is a priority. Despite their strong ability to render realistic images, it has been shown that GANs can learn biases that are present in the training dataset, which negatively impacts the network's ability to generate a diverse set of images. In particular, it has been shown that GANs trained on photos of centered human faces often fail to generate any meaningful images if the subject's face is not centered in the image frame. Prior work extended the StyleGAN2 architecture such that it was able to generate off-center images by adding a positional encoding element to the network. In this work, we further extend this spatially unbiased GAN to be able to predict the amount of translation an image has undergone, allowing the model to perform high-quality reconstructions *without* requiring users to provide the image translation parameters. Furthermore, we improve upon the original positional encoding capabilites to enable our network to generate images that have been subject to arbitrary affine transformations (e.g. rotations and shears). Note: This report contains new results that *were not* included in our final project presentation. These results can be found in Sections 6 and 7, which are headered with red text.

1. Introduction

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) are a powerful tool for generating highly realistic images in a variety of domains. Using GANs, we can learn a representation of a distribution without the need for large amounts of labeled data, and we can sample from this latent representation to produce new, realistic examples. Since our latent space is a representation of the input data distribution, it is common for biases present in the training data to also be present in the latent representation. In the case of a dataset of face images, prior work has shown that latent representations learned by GANs exhibit a strong spatial bias towards faces that are positioned at the center of the image (Choi et al., 2021).

One limitation of models that have learned a spatial bias is that they struggle to generate high-quality images for which this spatial bias is not present (e.g. a once-centered image which has been translated such that the person's face is no longer in the center of the image). In this work, we consider the capacity of GANs to generate high-quality images that do not include this spatial bias. Specifically, we investigate the efficacy of different methods for training GANs to generate images of faces where different affine transformations have been applied to the image (e.g. translations and shears). To this end, we have two main contributions that build upon prior work (Choi et al., 2021) to train spatially unbiased GANs. First, we extend the architecture to be able to predict the translation that the subject image has undergone. This functionality allows the model to perform faithful reconstructions during GAN inversion *without* requiring explicit translation information of the input image to. Second, we improve upon the the positional encoding functionality introduced by Choi et al. (Choi et al., 2021) by extending it to other affine image transformations. This extension allows our model to generate images that have undergone transformations other than translations (e.g. rotations and shears), which was not possible using the original model proposed in (Choi et al., 2021). Finally, we combine these methodologies to provide our model the ability to generate and perform GAN inversions of affine-transformed images without requiring the explicit transformation and without ever training on such affine-transformed images.

2. Related Work

2.1. Translation Invariance in Convolution Networks

When tasked with detecting objects in a frame, it is important to be able to identify objects irrespective of the objects' positions within the image frame in order to achieve robust detection rates. Although the convolution operation itself is translation-invariant, the convolutional neural network (CNN) architecture is *not* (Alsallakh et al., 2020; Azulay & Weiss, 2018; Kayhan & Gemert, 2020; Manfredi & Wang, 2020; Zhang, 2019). CNN models can learn translation-dependent biases from implicit position encoding from zero-padding (Islam et al., 2020; Kayhan & Gemert, 2020; Xu et al., 2021) and due to biases in datasets (Manfredi & Wang, 2020). Since it is clear that even models that are built upon

individually translation-invariant operations can exhibit spatial biases, like CNNs do, it is important that we develop specific techniques to train networks that are robust against spatial biases.

2.2. Positional Encoding

Positional encoding is a method for codifying the positions of elements within a sequence. Commonly used in natural language tasks (Vaswani et al., 2017), positional encoding provides an efficient way to integrate positional information into neural networks by adding a sinusoidal embedding to the input data. Positional encoding has also proven to be effective in the computer vision domain, yielding state-of-the-art performance at object detection (Carion et al., 2020) and segmentation (Caron et al., 2021) tasks. Additionally, positional encoding is a fundamental component in 3D image synthesis tasks, as demonstrated by the recent NeRF architecture (Mildenhall et al., 2020).

2.3. Learned Biases in Generative Models

When training generative models, it is common for the final model to have learned underlying biases in the training data (Esser et al., 2020; Zhao et al., 2018). If we want to develop models that are capable of generating a diverse set of images, it is important that our models have as little bias in them as possible, as this bias will constrain the models to generate only certain types of data. In addition to constraining the diversity of generated images, it has been shown that generative models that have learned biases in the data can use these biases as anchors during the generation process, which can serve as a "shortcut" which the model relies upon for faithful generation (Bahng et al., 2020). In this work, we focus on developing GANs that overcome their learned biases in the positions of faces within the image frame.

3. Original MS-PE and its Limitations

Multi-scale positional encoding (MS-PE) was introduced as a solution to the spatial bias inherent to many generative models. While current models are powerful, they have a strong bias towards images that are in positions seen during training; for example, if all images are centered, the model will be able to generate high-fidelity centered images, but will struggle with generating off-centered images. This is a clear flaw, as a model should be able to generate and recognize faces regardless of their positions in an image.

The solution presented in (Choi et al., 2021) was to borrow the idea of positional encoding from transformer literature, where the features for each pixel include information about its location in the image. Further, it was shown that single-scale positional encoding was ineffective, and that positional encoding was necessary at every scale of the generator (hence, multi-scale positional encoding).

Figure 1. Multi-scale positional encoding allows the StyleGAN2 architecture to perform GAN inversion of a translated image.

To encode a pixel's position, they use a continuous version of the binary representation of the x and y position, found using the following:

$$PE_{(i,j)} = \left[\sin(i/10000^{k/d}), \cos(i/10000^{k/d})_{k=0}^{d}\right] \in \mathbb{R}^{2d} \tag{1}$$

where $k = \{0, 1, ..., d-1\}$ and d is a quarter of the channel dimension, which allows us to add this encoding directly into our feature map at every scale. Finally, a learnable scalar is placed in front of the positional encoding in order to properly weight it as a feature.

By incorporating this technique into the StyleGAN2 (Karras et al., 2020) architecture, the authors showed that the resulting model was much more translation-invariant. Figure 1 depicts the result of a GAN inversion on a translated image by StyleGAN2 with and without MS-PE. We note that the only difference between models is the incorporation of MS-PE; both models were trained on centered images, but the model with MS-PE was able to generalize its knowledge to translated images. While an impressive result, we found two limitations of the current implementation of MS-PE that provided us with opportunities for improvement.

First, in order for MS-PE to work properly, it requires explicit transformation information. From a generation perspective, this is necessary to tell the model what kind of image to generate and can simply be randomized to generate an array of translated images. However, from a GAN inversion perspective, it is unrealistic that we would know the translation of an image prior to inversion. We hypothesized that this information could be estimated with high accuracy using neural networks. Second, the current implementation of MS-PE only supports transformations that are completely separable in x and y. We hypothesized that this was needlessly restrictive and that any affine transformation could be represented using a modified version of MS-PE.

Thus, our contributions are twofold. We first show that the

necessity of explicit transformation information restricts the model's ability to perform GAN inversion. To remedy this, we train a neural network that can accurately predict the translation information of an image, thus eliminating the need for the explicit transformation to perform accurate GAN inversions. Then, we modify the current MS-PE implementation to a more expressive version we call Affine MS-PE (and abbreviate as APE). We show that this implementation can allow a model to generate and invert certain affine-transformed images despite being trained on centered, untransformed images. Finally, we go on to combine both of these ideas to show that we can predict the transformation of an image and use this information to make a high quality reconstruction of an affine-transformed image.

For our experiments, we used a modified version of the Flickr-Face-HQ (FFHQ) dataset (Karras et al., 2019). The original dataset had images of resolution 1024×1024; we reduced the resolution of all images to 256×256 to facilitate faster computations. Further, the FFHQ dataset had $70,000$ images, of which we used the first $50,000$ for training and the next $1,000$ for testing. Unless otherwise specified, the models we train use this train/test split, and we train for 500 thousand iterations of SGD using a batch size of 8. We note that the original paper allowed for the generation of images at different resolutions by training on images of several resolutions; we neglect to do so in our experiments and only make use of 256×256 images. However, we expect our results to generalize to different resolutions when trained to do so, and leave this to future work.

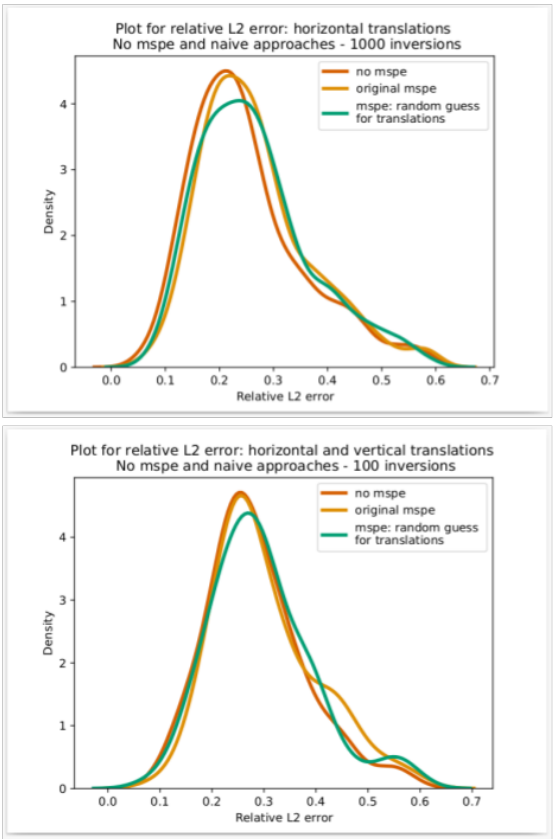

Figure 2. Comparison between the "Naive" MS-PE methods and classic StyleGAN2 (no MS-PE).

4. Learning Transformation Parameters

In this section we will show the motivation for our method based on learned transformations, then the results that we obtained on GAN inversion with translated images and comparisons to other methods.

4.1. Limitations of "Naive" MS-PE Methods

To motivate our experiments with learned transformations, we tried to understand how well the "naive" MS-PE methods perform when trying to do GAN inversions on translated images. We ran two experiments: 1) we applied GAN inversion on 1000 randomly horizontally translated images; 2) we applied GAN inversion on 100 randomly horizontally and vertically translated images. For both experiments, we compared three models: a) the classic StyleGAN2; b) StyleGAN2 with MS-PE and the assumption that all the images are centered; c) StyleGAN2 with MS-PE and using a random guess for the translation information. The metric we used is relative L2 reconstruction error. We present the results in Figure 2. From these experiments, we can conclude that the "Naive" MS-PE methods do not perform better than the classic StyleGAN2 (which does not use MS-PE at all).

4.2. Learning Transformations

We used a slightly modified AlexNet (Krizhevsky et al., 2012) architecture to predict the translation information: we replaced the last layer to have two output neurons and used the Sigmoid activation function. Note that later, we rescaled the output of the model to match the pixel translations (we multiplied by 255 and rounded the value to the closest integer). We did this because we found that the model learns better when the output is in [0, 1] compared to integers in [0, 255]. We trained the model for 100 epochs with a batch size of 512 and Adam optimizer (Kingma & Ba, 2014), with a learning rate of 0.001 reduced by a factor of 10 every 30 epochs. We ran an experiment similar to the one described in Section 4.1, but considering the following three methods: a) best "Naive" method, that is, classic StyleGAN2; b) MS-PE with the exact translation information known and c) ours, with the predicted translation information. The results are shown in Figure 3. The conclusion of this experiment is that our method outperforms the naive MS-PE approaches and performs almost as well as MS-PE with the translation

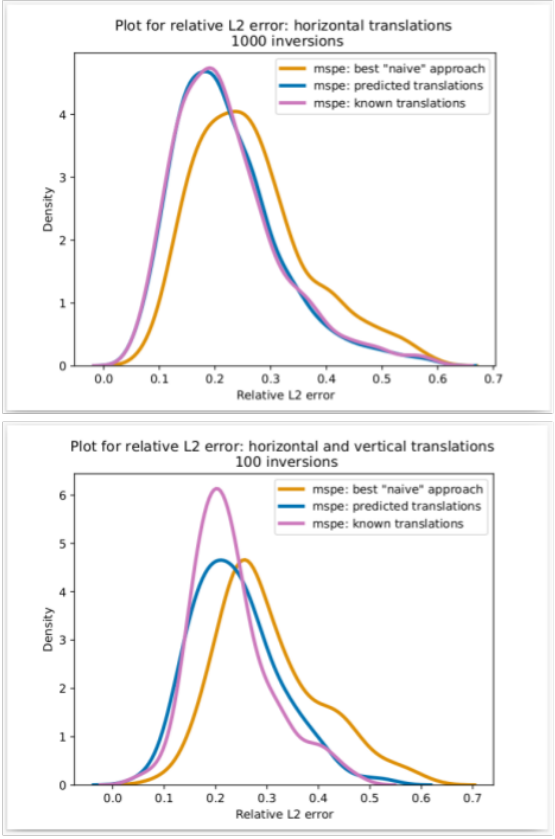

Figure 3. Comparison between our method, the best "Naive" MS-PE method and MS-PE with the exact translation parameters known.

information known. Also, in Figure 13 of Appendix A, there are some qualitative results for each method.

5. Affine MS-PE

The multi-scale positional encoding scheme used in (Choi et al., 2021) allows for translation-invariant GAN inversion, however real world images are often not only translated but photographically "shot" off of the z-axis (that is to say not head on). Recall that if the focal length of a camera is f then the "camera plane" projection of a point (x, y, z) in three dimensional space is $(fx/z, fy/z, f)$. This transformation is not linear (owing to the division by z) however it can be lifted to a linear transformation on the projective space $\mathrm{H}^3 = \mathbb{R}^4/\sim$ ($x \sim y$ if and only if there exists $\lambda \in \mathbb{R}$ so that $x = \lambda y$). In particular the camera coordinates are given by

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} fx/z \\ fy/z \\ f \\ 1 \end{bmatrix} \sim \begin{bmatrix} fx \\ fy \\ fz \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & f & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{Q} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2}$$

Thus if the camera is translated by $b \in \mathbb{R}^3$ and then rotated by $R \in SO(3)$ the resulting projective transformation of the coordinates in the camera plane is $\vec{x}' = QT_{\text{camera}}^{-1}\vec{x}$ where

$$T_{\text{camera}} = \begin{bmatrix} R & b \\ 0^T & 1 \end{bmatrix}. \tag{3}$$

Thus the projective Euclidean transformation T_{camera} acts on the image coordinates via the affine representation $\begin{bmatrix} A' & b' \\ 0^T & 1 \end{bmatrix}$ formed by removing the third row and column from $QT_{\text{camera}}^{-1}Q^\dagger$ ($Q^\dagger$ here is the Penrose pseudo-inverse of Q). All this to say that if we want to be able to perform generative tasks such as domain adaptation and GAN inversion on images that might be shot "off" of the z-axis, then it suffices to make our generator invariant to arbitrary affine transformations of $\mathbb{R}^2$, that is, the six parameter group:

$$\mathcal{A} = \{(A, b) | A \in \mathrm{GL}(2), b \in \mathbb{R}^2\} \tag{4}$$

where $(A, b) \cdot (A', b') = (AA', b + Ab')$ and the group acts on $\mathbb{R}^2$ via $(A, b)x = Ax + b = \begin{bmatrix} A & b \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix}$.

The qualitative results in Figure 4 indicate that MS-PE is *not* invariant to arbitrary affine transformations of the image, and hence will struggle with images not shot head on. Moreover, Figure 4 suggests that the "more extreme" the transformation the more MS-PE will deform the image. As it turns out, the reason for this failure is not a fundamental non-applicability of positional encoding to the affine case but a design choice in the positional encoding scheme used in (Choi et al., 2021) that implicitly assumes two features of the image transformation: In particular, (Choi et al., 2021) implicitly assumes that the image transformation commutes with the operation of taking the positional encoding and that it contains no "interaction" between x and y coordinates. The steps used for positional encoding at a particular scale in (Choi et al., 2021) are shown diagrammatically in Figure 5. Notice that with this approach it is not possible to transform a given row of the image's positional encoding in a way that depends on the row or column index (the simplest example of such a transformation is a shear, where the shift of a row is proportional to its index). Moreover, note that (cyclic) translation prior to positional encoding gives the same result

Figure 4. First row: Increasingly rotated target images. Second row: GAN projections of the rotated images. Third row: Increasingly sheared target images. Fourth row: GAN projections of the sheared images.

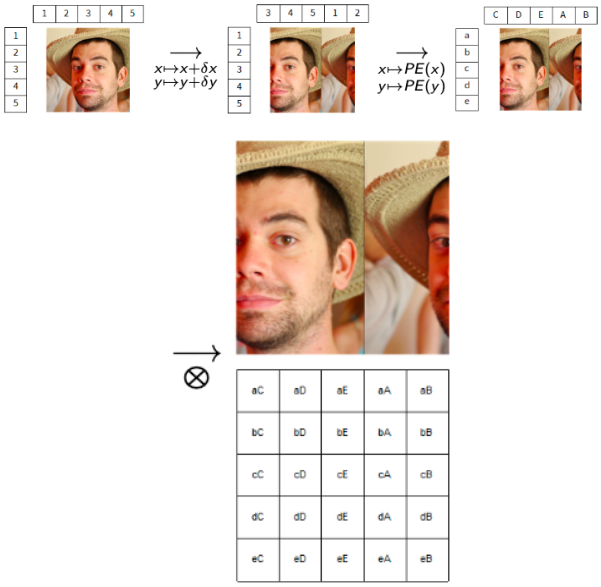

Figure 5. Step 1: Horizontal and vertical index vectors are separately transformed (in this case translated). Step 2: Horizontal and vertical index vectors are positionally encoded. Step 3: Image is positionally encoded via the the tensor product of the horizontal and vertical positional encodings.

as (cyclically) translating the positionally encoded vectors, making it irrelevant which occurs first. Together these two assumptions entirely restrict the applicability of MS-PE to translations – but they are not necessary!

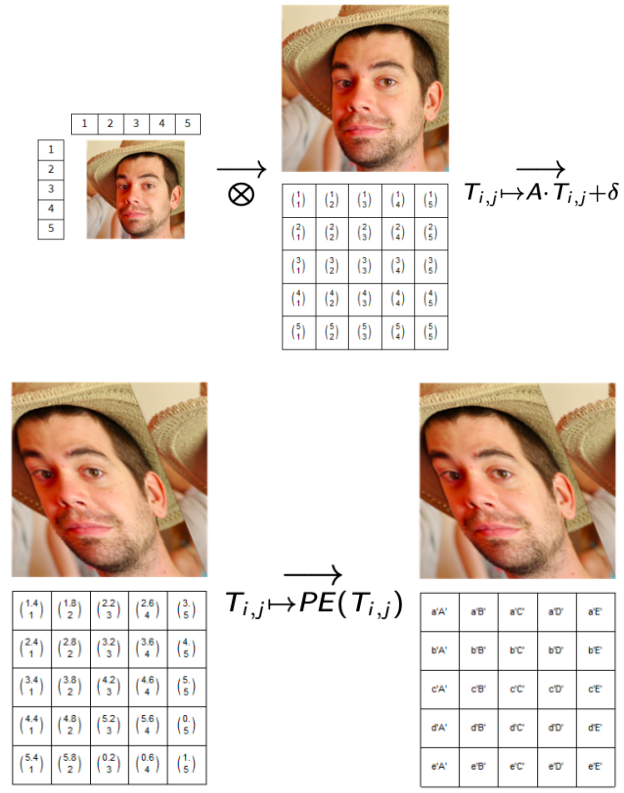

Figure 6. Step 1: Take the tensor product of the horizontal and vertical index vectors. Step 2: Transform the resulting index 3-tensor via a Hadamard action of the affine transformation. Step 3: Positionally encode the resulting transformed index tensor.

Indeed, positional encoding in general does *not* commute with affine transformations, and affine transformations require jointly transforming x and y coordinates. Our solution is the positional encoding scheme laid out in Figure 6. Under this scheme, we form a joint index based positional encoding tensor first, apply the desired transformation via a Hadamard action (entry-wise along the first and second tensor indices), and only then do we form the continuous binary positional encoding tensor (again in an entry-wise fashion). This scheme agrees with that in (Choi et al., 2021) if the transformation acts separately on x and y and commutes with positional encoding. A key point here is that this positional encoding also acts on multiple scales; however, unlike translation, the linear part of an affine transformation "looks the same at every scale" and does not require re-scaling (this property could be taken as a heuristic definition for linear transformations).

Quantitative results may be found in Figures 7 and 8 for

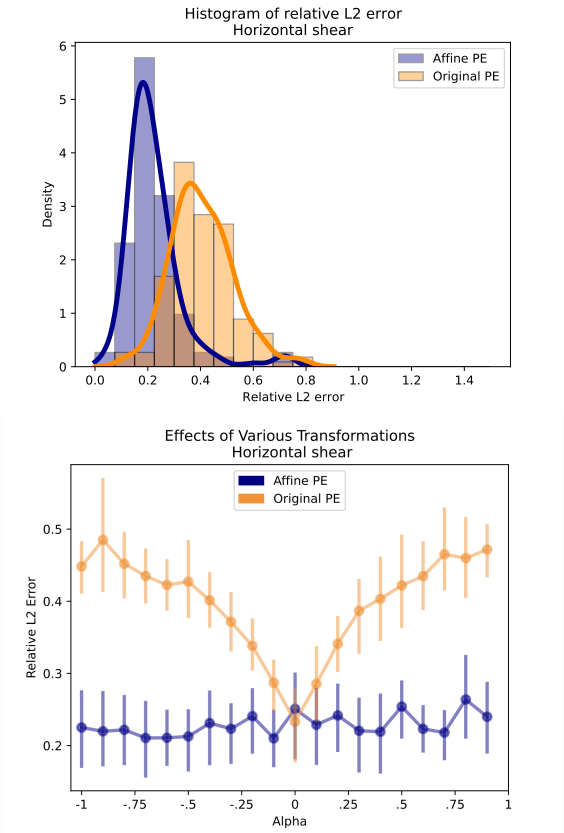

Figure 7. Top: Histogram of l^2 reconstruction error for 50 images with three different shears applied to each. Bottom: l^2 reconstruction error as a function of shear parameter α for five images and 20 discrete angles.

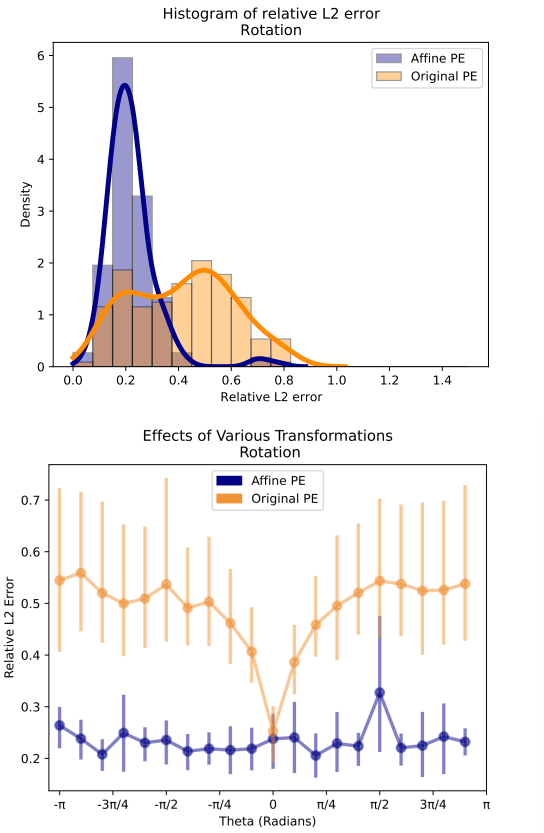

Figure 8. Top: Histogram of l^2 reconstruction error for 50 images with three different rotations applied to each. Bottom: l^2 reconstruction error as a function of rotation angle θ for five images and 20 discrete angles.

shears and rotations (qualitative results can be found in the appendix). As can clearly be seen, our Affine Positional Encoding (APE) performs as well as the original positional encoding scheme when no transformation is applied to the image, but as we increase the distance of the transformation from identity (via angle θ and shear parameter α respectively) our method outperforms MS-PE, and indeed appears not to suffer much in performance at all. We also analyzed the relationship between reconstruction error and $||A - \mathbb{I}||_F$ directly (here A is the linear part of the affine transformation) for 150 random linear transformations (each of the four matrix entries sampled uniformly from $[-1, 1]$). The results can be seen in Figure 9, confirming that our method improves on (Choi et al., 2021) not only for rotation and shearing but for arbitrary linear transformations.

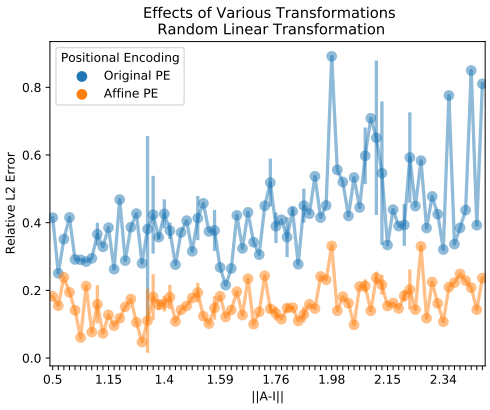

Figure 9. l^2 reconstruction error as a function of $||A - \mathbb{I}||_2$ for 150 entry-wise uniformly random transformations.

6. Comparing our Methods to StyleGAN2 Trained on Transformed Images

In some sense, it is unfair to expect a standard StyleGAN2 architecture to be able to perform GAN inversions of transformed images when it is trained on aligned, centered images. This sections attempts to provide a a fairer comparison between StyleGAN2 and APE. To do so, we train a StyleGAN2 model on translated and rotated images and compare

its inversions to a model that makes use of APE. We expect that APE outperforms the original StyleGAN2 while also encompassing a wider range of transformations than translations and rotations.

For the translation experiment, we trained a model on 50000 images that were vertically and horizontally shifted by an integer number of pixels from 0 to 255 and compared to our model that uses APE and predicts the translation information. We performed GAN inversion on 1000 images that were shifted only horizontally and 100 images translated vertically and horizontally. Figure 10 shows the reconstruction errors; in both cases, APE with predicted translations outperforms the original StyleGAN2.

Similarly, we went on to train a model on 50000 images, where images were rotated by 0, 90, 180, and 270 degrees. We then inverted 1000 images rotated by a multiple of 90 degrees and compared to our model with APE. Figure 11 depicts the reconstruction errors, where we can see that APE outperforms StyleGAN2. We note that, for this experiment, we assumed that the rotations were known for the APE model; however, Section 7 provides evidence that this assumption likely can be removed. We also note that APE supports a much wider class of transformations than rotations, making it a strictly better method than attempting to enumerate all possible transformations and train on them.

7. Predicting Affine Transformations

In this section, we consider the following experiment: we compare APE with predicted values for the 6 parameters of an affine transformation, Original (centered) MS-PE and APE with the 6 parameters of the affine transformation known. We used randomly sampled values in the range $(-50, 50)$ for the two translation parameters and random samples in the l^∞ ball of radius 1 around the 2×2 identity matrix for the other four parameters. To predict the 6 parameters, we used an AlexNet model similar to the one in Section 4.2, but with Tanh activation instead of Sigmoid and reducing the learning rate every 40 epochs instead of 30. Similarly, we rescaled the output to match the targeted range, where necessary ((-50, 50) or (0, 2)).

We ran GAN inversion with 100 samples and shown the plots in Figure 12. The results show that APE with predicted affine parameters performs significantly better than Original MS-PE and almost as good as APE with known affine parameters. We included some qualitative results in Appendix C, Figure 17.

8. Limitations and Future Work

So far, most of the experiments we describe have shown success for APE and the ability to predict transformations to

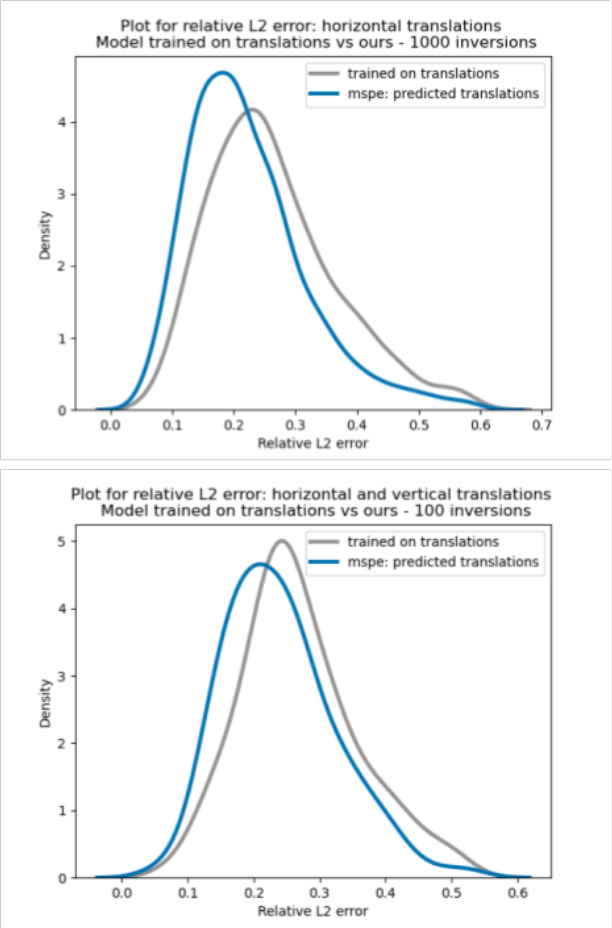

Figure 10. Comparison between our method (predicted translations) and StyleGAN2 trained on translated images.

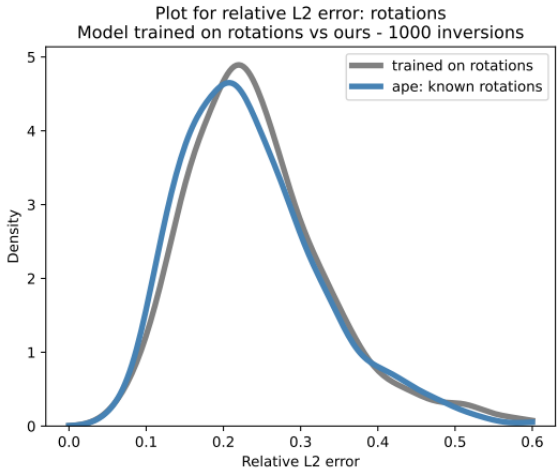

Figure 11. Comparison between our method (Affine MS-PE with known transformations) and StyleGAN2 trained on rotated images.

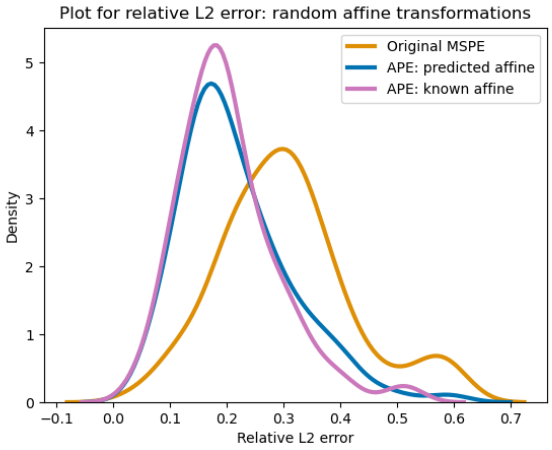

Figure 12. Comparison between APE with predicted parameters, APE with known parameters and Original MS-PE.

facilitate GAN inversion. It is not always the case that APE outperforms MS-PE during inversion; for example, row 3 of Figure 17 in Appendix C shows an example of an affine-transformed image where MS-PE creates a better inversion. However, we claim, that in most cases, APE should perform better during inversion of affine-transformed images, even when we must predict the transformations ourselves.

While we argue that the ability to perform GAN inversion makes our methods a success, GANs are more commonly used for image generation rather than inversion. Figure 15 in Appendix B shows *generated* rotated images via APE. This is a task (unlike inversion) that the original positional encoding method cannot even attempt, as there is no way to tell it what transformation to apply. Interestingly, APE does not do very well once the rotation angle exceeds 90 degrees – it blurs the image. This should not be that surprising, since the network was not trained on rotated images and hence, while it knows the approximate positions of things via APE it cannot fill in the details. Thus, it seems likely that, during GAN projection, APE provides coarse grained positional information that is lacking with the positional encoding scheme in (Choi et al., 2021), while the single image being projected provides the fine grained information to fill in the details.

In terms of future work, we have shown that our work generalizes the space of images a generative model can generate and invert, assuming that they are of the resolution 256x256. This is the only aspect of the original MS-PE paper we have not expanded upon, wherein they train a model on several different resolutions and show their model can generate images of arbitrary scale. Thus, the first area of future work should be to show that our methods work on images of different resolutions, which would confirm our methods are strictly more representative than those of the original paper.

With the introduction of an additional neural network to predict transformation information, a new front of vulnerability has been introduced to the model; an adversary could now attack the original GAN architecture or the model predicting transformations during GAN inversion. We have no reason to believe Affine MS-PE would make the GAN more vulnerable, but the same cannot be said for the transformation predictor. We made no guarantees for this model under adversarial attacks, and future work should investigate if this is a potential vulnerability and, if so, how to make it robust.

Finally, our experiments were limited to "reasonable" affine transformations. As images get transformed more aggressively, we would expect the image quality to degrade. It remains to be shown how far we can push the assumption of "reasonable" affine transformations and if the assumption is necessary at all. Future work should investigate the empirical and theoretical bounds on the range of affine transformations Affine MS-PE can handle. Further, a method of quantifying how much a transformation distorts an image is vital. We hypothesize that the Frobenius norm between the 2x2 transformation matrix (i.e. an affine transformation with no translation) and the identity matrix would be a good measure, as evidenced by Figure 9.

9. Conclusion

In this work, we investigated the ability for generative adversarial networks (GANs) to be able to synthesize realistic images of humans faces that have undergone arbitrary affine transformations. Building upon work by Choi et al. (Choi et al., 2021), we developed a model capable of predicting the vertical and horizontal translations that an image has undergone. This model lead to improved performance in generating translated images *without* requiring explicit translation information to be fed into the input. Additionally, we improved the model's ability to learn positional information in images by extending the positional encoding formulation to work with *arbitrary* affine transformations. Using our improved Affine MS-PE, our model was capable of generating realistic images of human faces that have undergone more complex transformations such as rotations and shears. With these improvements, we demonstrated that there is room for improvement in the ability of GANs to be positionally unbiased, and we demonstrated two ways in which such biases can be mitigated. Future work should investigate the ability of GANs to generate realistic images that have undergone 3D transformations.

References

Alsallakh, B., Kokhlikyan, N., Miglani, V., Yuan, J., and Reblitz-Richardson, O. Mind the pad–cnns can develop blind spots. *arXiv preprint arXiv:2010.02178*, 2020.

Azulay, A. and Weiss, Y. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018.

Bahng, H., Chun, S., Yun, S., Choo, J., and Oh, S. J. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pp. 528–539. PMLR, 2020.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pp. 213–229. Springer, 2020.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.

Choi, J., Lee, J., Jeong, Y., and Yoon, S. Toward spatially unbiased generative models. *arXiv preprint arXiv:2108.01285*, 2021.

Esser, P., Rombach, R., and Ommer, B. A note on data biases in generative models. *arXiv preprint arXiv:2012.02516*, 2020.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Islam, M. A., Jia, S., and Bruce, N. D. How much position information do convolutional neural networks encode? *arXiv preprint arXiv:2001.08248*, 2020.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks, 2019.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, 2020.

Kayhan, O. S. and Gemert, J. C. v. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14274–14285, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105, 2012.

Manfredi, M. and Wang, Y. Shift equivariance in object detection. In *European Conference on Computer Vision*, pp. 32–45. Springer, 2020.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pp. 405–421. Springer, 2020.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Xu, R., Wang, X., Chen, K., Zhou, B., and Loy, C. C. Positional encoding as spatial inductive bias in gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13569–13578, 2021.

Zhang, R. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pp. 7324–7334. PMLR, 2019.

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. Bias and generalization in deep generative models: An empirical study. *arXiv preprint arXiv:1811.03259*, 2018.

A. Qualitative Results for Translations Experiments

In Figures 13 and 14 we show some qualitative results for the translations experiments.

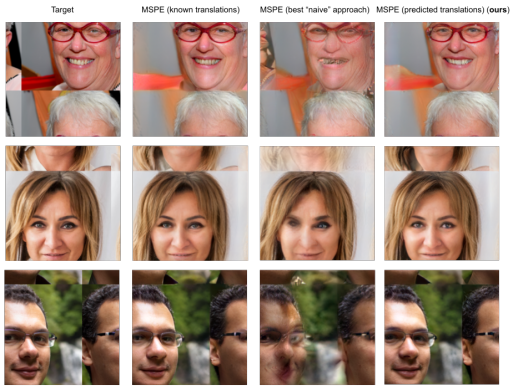

Figure 13. The first column represents the target, the second one for the experiment with the translation parameters known, the third one is the best "Naive" method and the last one is ours.

Figure 14. Qualitative results for StyleGAN2 trained on translated images and our method (with predicted translations).

B. Qualitative Results for Shear and Rotation Experiments

Figure 15 shows *generated* rotated images via APE. Figure 16 shows qualitative results of APE vs MS-PE for rotations and shears.

Figure 15. Generated rotated images using APE.

C. Qualitative Results for Random Affine Transformations

Figure 17 shows qualitative results in GAN inversion with random affine transformations applied to the samples.

Figure 16. Qualitative results of APE vs MS-PE. While our method is not perfect for more extreme transformations, it nevertheless significantly outperforms MS-PE.

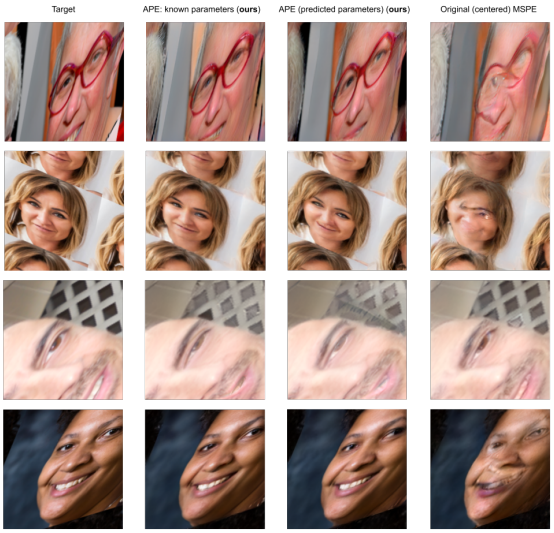

Figure 17. Qualitative results for random affine transformations APE (known or predicted parameters) vs Original (centered) MS-PE.