# Alias-Free Convnets: Fractional Shift Invariance via Polynomial Activations

Hagay Michaeli  Tomer Michaeli  Daniel Soudry
Department of Electrical and Computer Engineering, Technion
Haifa, Israel
{hagaymichaeli, daniel.soudry}@gmail.com, tomer.m@ee.technion.ac.il

## Abstract

*Although CNNs are believed to be invariant to translations, recent works have shown this is not the case due to aliasing effects that stem from down-sampling layers. The existing architectural solutions to prevent the aliasing effects are partial since they do not solve those effects that originate in non-linearities. We propose an extended anti-aliasing method that tackles both down-sampling and non-linear layers, thus creating truly alias-free, shift-invariant CNNs*[1]*. We show that the presented model is invariant to integer as well as fractional (i.e., sub-pixel) translations, thus outperforming other shift-invariant methods in terms of robustness to adversarial translations.*

## 1. Introduction

Convolutional Neural Networks (CNNs) are the most common model in the image classification field. They were originally intended to have two properties:

1. Shift-invariant output: when we spatially translate the input image, their output does not change.
2. Shift-equivariant representation: when we spatially translate the input image, their internal representation translates in the same way.

Both these properties are thought to be beneficial for generalization (*i.e.*, they are useful inductive biases), as we expect the image class not to change by an image translation, and its features to shift together with the image. Moreover, without the first property, the CNN might become vulnerable to adversarial attacks using image translations. Such attacks are real threats since they are very simple to execute in a "black-box" setting (where we do not know anything about the CNN). For example, consider a person trying to fool a CNN-based face scanner, by simply moving continuously until a face match is achieved.

It was commonly assumed that these useful properties were maintained since CNNs use only shift-equivariant operations: the convolution operation and component-wise non-linearities. However, CNN models typically also include downsampling operations such as pooling and strided convolution. Unfortunately, these operations violate equivariance, and this also leads to CNNs not being shift-invariant. Specifically, Azulay and Weiss [2] have shown that shifting an input image by even one pixel can cause the output probability of a trained classifier to change significantly. This vulnerability can be further exploited in adversarial attacks, lowering classifiers' accuracy by more than 20% [8]. Later, Zhang [33] has shown that this problematic behavior stems from an aliasing effect, taking place in downsampling operations such as pooling and strided convolutions, and non-linear operations on the downsampled signals.

Previous works have shown an improvement in CNN invariance to translations using partial solutions that reduced aliasing. For example, Zhang [33] has suggested adding a low-pass filter before the downsampling operations. This approach has been shown to reduce aliasing caused by downsampling, thus improving shift-invariance, as well as accuracy and noise robustness. Karras *et al*. [17] have addressed aliasing in the generator within generative adversarial networks (GANs). They have shown that without proper treatment, aliasing in GANs leads to a decoupling of the high-frequency features (texture) from the low-frequency content (structure) in the generated images, thus limiting their applicability in smooth video generation. To alleviate this issue, Karras *et al*. [17] extended the low-pass filter approach and suggested a solution for the implicit aliasing caused by non-linearities. Their method wraps the component-wise non-linear operations by upsampling and downsampling layers in an attempt to mimic the effect of applying the non-linear operations in the continuous domain, where they theoretically do not cause aliasing.

Yet, none of the previous solutions completely eliminates aliasing, thus their suggested CNN architectures are not guaranteed to be shift-invariant. A different approach to shift-invariant CNNs was suggested by Chaman and Dokmanić [4]. They have proposed to use downsampling operations that dynamically choose the subsampling grid using

[1] Our code is available at github.com/hmichaeli/alias_free_convnets/.

a shift-equivariant decision rule. Although it does not solve the aliasing problem, nor guarantees shift-equivariant representations, this approach enables the creation of CNNs whose outputs are completely invariant to integer circular shifts. However, this approach does not lead to invariance to subpixel shifts, which are common in real-world applications. For example, consider a case where the CNN receives input from a camera with some finite resolution. If we continuously shift the camera with respect to the scene, then the resulting shift in the CNN's discretely sampled input would rarely be integer-valued.

Considering the anti-aliasing approach again, the problem with the solution suggested by Karras *et al*. [17] is that the aliasing resulting from non-linearities that increase the signal's bandwidth indefinitely (such as ReLU) can be avoided only when they are used in a continuous domain (*i.e*., with infinite resolution), which is impractical. However, this problem can be solved by replacing such non-linearities with alternatives whose effect does not lead to an indefinite increase in the signal's bandwidth — such as polynomials.

**Polynomial activations** Despite their ease of computation, polynomials are not considered promising candidates for activation functions. The main practical reason for this is that polynomial activations have large (super-linear) magnitudes compared to standard activations (*e.g*., ReLU) and thus typically cause training instability (*e.g*., exploding gradients) [11]. There seems also to be a theoretical disadvantage since shallow feedforward neural networks with polynomial activation functions are not universal approximators [15]. However, this last issue may not be a serious disadvantage: Kidger and Lyons [18] have shown that feedforward neural networks with polynomial activations can become universal approximators with sufficient depth — a regime more relevant for modern CNNs. In addition, recent research [11] has shown that by using normalization to truncate the dynamic range of the pre-activations, the training of Neural Networks with polynomial activations can be stabilized, and converge to reasonable results in simple image classification tasks (MNIST and CIFAR). Yet, there are still a few significant challenges in using polynomial activations: First, to the best of our knowledge, they were not shown to achieve competitive performance (similar to standard activations) on tasks of more realistic scales, such as ImageNet. In addition, the normalization method for dynamic range truncation causes the (truncated) polynomial to increase the signal's bandwidth indefinitely, which is not suitable for aliasing-free CNNs. This normalization was shown to be crucial for convergence even in small tasks and it is reasonable to expect that is even more important for larger tasks.

**Contributions** In this paper

- We propose the first Alias-Free Convnet (AFC).
- We prove the AFC has both shift-invariant outputs and shift-equivariant internal representations — even for fractional shifts, where previous models fail.
- We show how simple and easy "black-box" adversarial attacks built on fractional image translation can degrade a CNN performance, even when the CNN is invariant to integer shifts. In contrast, the AFC has certified robustness to such attacks and superior test accuracy in this regime.
- Specifically, the robustness of AFCs is certified for circular shifts and the ideal (Sinc) interpolation kernel. However, we show empirically that AFCs have improved robustness even with other types of translations.
- Interestingly, our model relies on polynomial activations, and we are the first to demonstrate competitive performance with such activations on ImageNet, to the best of our knowledge.

## 2. Methods

Let $\tau_\Delta : L^2(\mathbb{R}^2) \to L^2(\mathbb{R}^2)$ be the translation operator, which shifts a continuous-domain two-dimensional signal by $\Delta \in \mathbb{R}^2$. An operator $f : L^2(\mathbb{R}^2) \to L^2(\mathbb{R}^2)$ is said to be *shift-equivariant* if it commutes with $\tau_\Delta$ for every $\Delta$. Namely,

$$f\left(\tau_\Delta\left(x\right)\right) = \tau_\Delta\left(f\left(x\right)\right)$$

for every $x \in L^2(\mathbb{R}^2)$ and every $\Delta \in \mathbb{R}^2$.
An operator $f : L^2(\mathbb{R}^2) \to \mathbb{R}^d$ is said to be *shift-invariant* if its output is invariant to translation of its input, *i.e*.

$$f\left(\tau_\Delta\left(x\right)\right) = f\left(x\right) .$$

The definitions of equivariance and invariance to translations naturally transfer to discrete-domain signals in $L^2(\mathbb{Z}^2)$ and integer shifts $\Delta \in \mathbb{Z}^2$. To simplify notations, from now on we will not specify the domain over which operators are defined, and will also omit the subscript $\Delta$ from $\tau$, whenever the meaning is clear from the context.

CNN architectures for classification commonly comprise a *Feature Extractor*, which is mainly composed of convolution layers, and a *Classifier*, which is typically composed of a linear layer and a softmax activation. A sufficient condition for the model to be shift-invariant is that the *Classifier* be shift-invariant, and the *Feature Extractor* be shift-equivariant. This is because the composition of a shift-equivariant $f$ and a shift-invariant $g$ yields a shift-invariant function, as

$$g\left(f\left(\tau\left(x\right)\right)\right) = g\left(\tau\left(f\left(x\right)\right)\right) = g\left(f\left(x\right)\right) .$$

For our discussion, we assume that the *Classifier* is shift-invariant as its inputs are the spatially-averaged channels. However, the *Feature Extractor* part of CNNs commonly

includes also downsampling layers. The spatial dimensions of the output of such layers are smaller than the spatial dimensions of their input. Therefore, for such layers, shift-equivariance is not a desired property. Indeed, when shifting an image by 2 pixels at the input of a layer that performs downsampling by a factor of 2, we expect the output image to shift by only 1 pixel, not 2. Even worse, when shifting an image by only 1 pixel, it is not clear how precisely the output should shift. In order to extend the discussion to include these networks, here we consider equivariance w.r.t. the continuous domain. To simplify the exposition, let us present the definitions for 1D signals, where 'discrete' and 'continuous' will refer to the signal index we use. Namely, a discrete signal $x[n]$ is defined over $n \in \mathbb{Z}$ while a continuous signal $x(t)$ is defined over $t \in \mathbb{R}$.

**Definition 1 (Fractional translation for discrete signals)** *Let $x[n]$ be a discrete-domain signal and let $\Delta \in \mathbb{R}$ be a (possibly non-integer) shift. Then the translation operator $\tau_\Delta$ is defined by $\tau_\Delta(x)[n] = z(nT + \Delta)$, where $z(t)$ is the unique $1/2T$-bandlimitted continuous-domain signal satisfying $x[n] = z(nT)$.*

Note that the uniqueness of $z(t)$ in Definition 1 is guaranteed by the Nyquist theorem. It is also easily verified that this definition does not depend on $T$. Equipped with this definition, we can define the following.

**Definition 2 (shift-equivariance w.r.t. the cont. domain)** *An operator $f$ operating on discrete signals is said to be shift-equivariant w.r.t. the continuous domain if it commutes with fractional shifts. Namely, $f(\tau_\Delta(x)) = \tau_\Delta(f(x))$ for every $x \in L^2(\mathbb{Z})$ and every $\Delta \in \mathbb{R}$.*

Similarly, we can define the following.

**Definition 3 (shift-invariance w.r.t. the cont. domain)** *An operator $f$ operating on discrete signals is said to be shift-invariant w.r.t. the continuous domain if it is invariant to fractional shifts of its input. Namely, $f(\tau_\Delta(x)) = f(x)$ for every $x \in L^2(\mathbb{Z})$ and every $\Delta \in \mathbb{R}$.*

An important observation is the following.

**Proposition 1** *In a network comprised of a Feature Extractor and a Classifier, if the Feature Extractor ends with a global average pooling layer, then shift-equivariance w.r.t. the continuous domain of the Feature Extractor implies shift-invariance w.r.t. the continuous domain of the entire model.*

Indeed, in this case, the Classifier's input is only dependent on the average of the Feature Extractor, which is shift-invariant. The last statement stems from the fact that when shifting the input of an operator that is shift-equivariant w.r.t. the continuous domain, the output must be a faithful translated discrete representation of the same continuous signal. Namely, there exists some $1/2T$-bandlimited continuous signal $\tilde{f}(t)$ such that $f(x)[n] = \tilde{f}(nT)$ and $f(\tau(x))[n] = \tilde{f}(nT + \Delta)$. Thus, the averages of $f(x)[n]$ and $f(\tau(x))[n]$ are both equal to the "DC component" of $\tilde{f}$, and therefore must be equal.

In order to examine the property of equivariance w.r.t. continuous domain of CNNs, we shall look at the discrete signal that propagates in a CNN as a representation of a continuous signal, and at each layer as a representation of a continuous operation on the continuous signal. As shown by Karras *et al*. [17], aliasing in the discrete representation prevents shift-invariance of CNNs since it decouples the discrete signal from its continuous equivalent. In contrast, they have shown that alias-free operations preserve shift-equivariance w.r.t. continuous domain, and lead to shift-invariant CNNs. There, Karras *et al*. [17] have shown that convolutions and downsamplers which are properly treated using low-pass filters (LPFs), are indeed alias-free and thus shift-equivariant w.r.t. the continuous domain. In addition, they proposed a method to reduce the implicit aliasing of non-linearities which we describe next.

In the continuous domain, pointwise non-linearities may induce indefinitely high new frequencies. Applying a pointwise non-linearity in the discrete domain is equivalent to sampling a continuous signal after applying the pointwise non-linearity — which may break the Nyquist condition and cause aliasing. This implies that pointwise nonlinearities applied in the discrete domain are generally not shift-invariant w.r.t. the continuous domain. Using upsampling before the non-linearity may solve this problem since it increases the frequency support that does not cause aliasing. However, this approach cannot generally prevent aliasing, since the new frequencies generated by non-linear operations can be arbitrarily high. For example, the outputs of non-differentiable operations such as ReLU can have infinite support in the frequency domain, thus aliasing will be induced for every finite upsampling factor.

In this study, we propose replacing non-linear operations with a band-limited preserving alternative — polynomial functions. The proposed scheme for an aliasing-free polynomial function of degree $d$ is defined in Algorithm 1. In this algorithm, $\mathrm{Upsample}_z$ performs upsampling by a factor $z$ (*i.e*. resampling the input continuous signal at a $z\times$ larger sampling frequency), $\mathrm{LPF}_z$ is an ideal low-pass filter with cut-off $z$, $\mathrm{Downsample}_z$ performs downsampling by a factor $z$ (*i.e*. dividing the sample frequency by $z$), and

$$\mathrm{Poly}_d(x) = \sum_{i=0}^{d} a_i x^i \,. \tag{1}$$

The practical implementations of the operations above are described in Section 3 and Appendix C. Our contribution to the general framework that has been presented by Karras *et al*. [17] is the usage of polynomial activations,

**Algorithm 1** Alias-free polynomial activation

**Inputs:** $x$ - input signal, $\text{Poly}_d$ - polynomial of degree $d$.

$x_{\text{up}} \leftarrow \text{Upsample}_{\frac{d+1}{2}}(x)$

$y_{\text{poly}} \leftarrow \text{Poly}_d(x_{\text{up}})$

$y_{\text{LPF}} \leftarrow \text{LPF}_{\frac{2}{d+1}}(y_{\text{poly}})$

$y \leftarrow \text{Downsample}_{\frac{d+1}{2}}(y_{\text{LPF}})$

**Output:** $y$

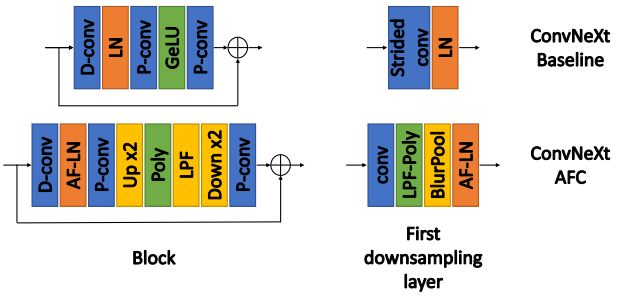


Figure 1. **ConvNeXt baseline architecture vs AFC modifications.** D-conv: depthwise convolution $7 \times 7$, P-Conv: pointwise convolution, Strided-conv: convolution $4 \times 4$, stride 4. LN: Layer Norm, AF-LN: Alias free Layer Norm, Poly: Polynomial activation. Up x2: Upsample x2, LPF: ideal LPF with cutoff 0.5, Down x2: Downsample x2. Detailed explanations about BlurPool, Poly and LPF-Poly activations can be found in Section 3.

which extends the frequency bandwidth in a limited fashion, unlike other non-linearities. Hence, by using appropriate upsampling as in Algorithm 1, aliasing can be avoided, as described in Figure 2. Specifically, in Appendix F.1 we prove the following.

**Proposition 2** *The operator defined by Algorithm 1 is shift-equivariant w.r.t. the continuous domain.*

By combining Proposition 1 and Proposition 2 with the shift-equivariance of the other layers (as described above), we conclude the network output is shift-invariant. Next, we describe the proposed process of non-linearities in the frequency domain, which is additionally demonstrated in Figure 2. In the first step, the input $x$ is upsampled, leading to a contraction of its support in the frequency domain (Fig. 2(b)). Effectively, it expands the range of allowed new frequencies generated by the following non-linearity (Fig. 2(c1)). Then, a low-pass filter is applied in order to prevent aliasing in the following downsampling layer (Fig. 2(d1), (e1)). Overall, for an upsampling factor that is appropriate for the frequency expansion of the polynomial, the effective frequencies for the output are not being overlapped at any of the steps, thus aliasing is prevented. However, in the case of non-linearities that do not preserve the band-limited property, upsampling cannot prevent the frequency overlap in Figure 2(c2).

## 3. Implementation

We propose an Alias-Free Convnet (AFC), based on the ConvNeXt architecture [19], which has been shown to achieve state-of-the-art results in image classification tasks. We modify the layers which suffer from aliasing (as described in Fig. 1) so that the convnet is completely free of aliasing. The theoretical derivation in Section 2 assumes infinite-length discrete signals, hence cannot be directly applied in practical systems. However, it can be naturally used by limiting the discussion to circular translations, which implies that the continuous signals are periodic. In this case, the theoretical results from Section 2 can be equivalently attained with finite-length signals using our following implementation.

**Convolution** We use circular convolutions to meet the periodic signal assumption, as described above. This is practically done by replacing zero padding with circular padding, similarly to Chaman and Dokmanić [4].

**BlurPool** Similarly to the model presented by Zhang [33], we separate strided convolutions into linear convolution and downsampling operations. The downsampling operation is replaced by BlurPool, which applies sub-sampling after low-pass filtering. Instead of implementing a low-pass filter using convolutions with custom fixed kernels, we implement an "ideal low-pass filter" by truncating high frequencies in the Fourier domain. Specifically, we transform the input to the Fourier domain using Pytorch FFT kernel [23], zero out the relevant frequencies, and transform it back to the spatial domain. This is an efficient implementation of downsampling after applying multiplication with filter $H^{2D}$ in DFT domain, which is defined as

$$H^{2D} = HH^T\,, \tag{2}$$

where for stride $s$ and spatial-domain size $N \times N$ , $H$ is defined as

$$H[k] = \begin{cases} 1, & 0 \le k < \frac{N}{2s}\,, \\ 0, & \frac{N}{2s} \le k \le \frac{3N}{2s}\,, \\ 1, & \frac{3N}{2s} < k \le N-1\,. \end{cases} \tag{3}$$

A derivation of this filter can be found in Appendix G.

**Activation function** We replace the original GeLU activation with a polynomial function of degree 2, whose coefficients are trainable parameters, per channel:

$$\text{Poly}_2(x) = a_0 + a_1 x + a_2 x^2\,. \tag{4}$$

The coefficients $\{a_0, a_1, a_2\}$ are initialized by fitting this function to the GeLU, as proposed by Gottemukkula [11]. All activation functions are wrapped according to the alias-free technique presented in Algorithm 1. Generally, replacing the activation function in a Deep Neural Network may

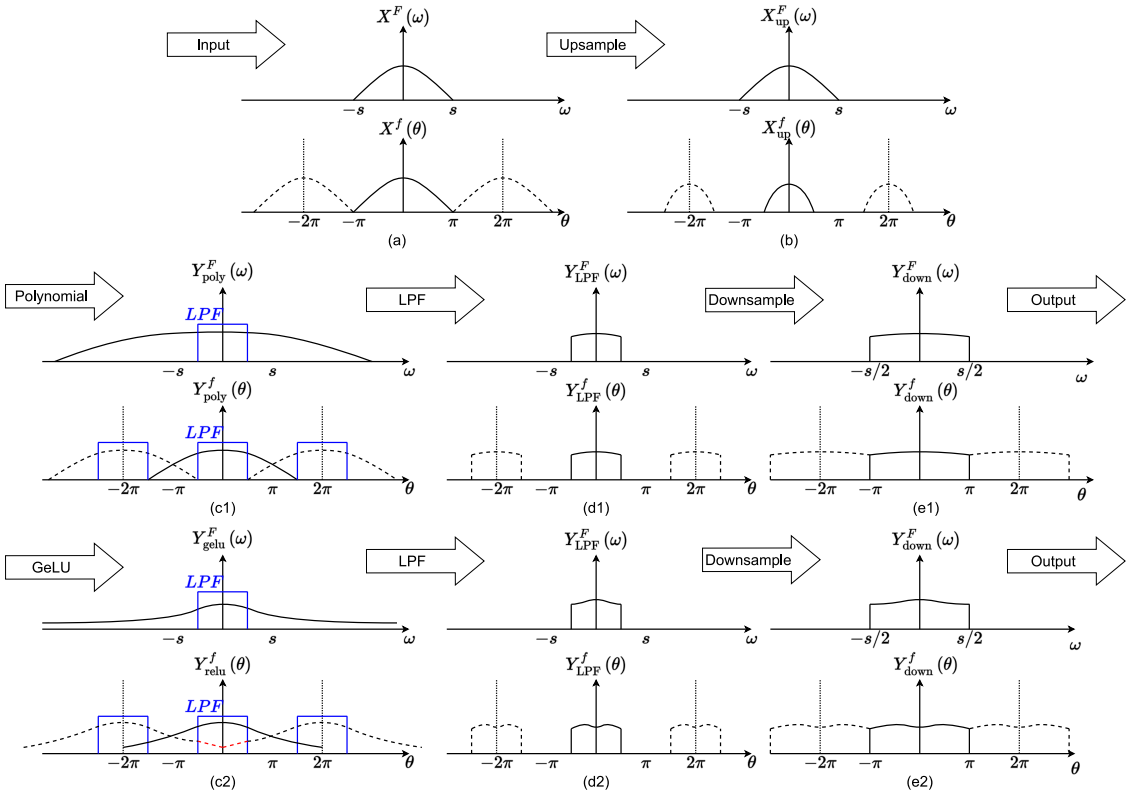


Figure 2. **A demonstration of the proposed non-linearities in the frequency domain.** The top plot at each panel represents the signal in the continuous domain, and the bottom represents the discrete domain. Where the input (a) is upsampled it shrinks its frequency response, expanding the allowed frequencies (b). Applying the polynomial activation expands the frequency response support by as factor $d$, without causing aliasing in the relevant frequencies (c1). Thus, the discrete signal remains a faithful representation of the continuous signal after applying LPF (d1) and downsample back to the same spatial size (d2). However, applying GeLU expands the support infinitely (c2). This leads to an aliasing effect — interference in the relevant frequencies marked in red in (c2). This causes the discrete signal not to be a correct representation of the continuous one, after LPF (d2) and downsampling (e2).

change the scale of the propagated activation, thus requiring adjusting the weight initialization. In our experiments the activation scale had a large impact on the achieved accuracy, thus searching for an appropriate scale factor was required. Details regarding the activation tuning can be found in Appendix D. Overall, in our case (polynomial activation in ConvNeXt) using the appropriate scale seemed to recover most or all of the lost accuracy.

**Normalization** ConvNeXt model implementation uses a variation of LayerNorm, which centers and scales each pixel according to its mean and standard deviation over channels, respectively. The scaling operation requires the multiplication of each pixel with a different scalar which, like other point-wise non-linearities, is not alias-free. We construct an alias-free alternative by using scaling per layer instead of scaling per pixel, *i.e*. all pixels are scaled by the standard deviation of the layer, which is shift-equivariant w.r.t. the continuous domain. Although eliminating aliasing effects, this modification caused a small reduction in the model accuracy, as we shall see later. We hypothesize this reduction results from the "normalization-per-pixel" operation functioning as an additional non-linearity, which enlarges the model capacity. Yet, this modification is required for the property of shit-equivariance w.r.t. continuous domain, which leads to an overall improvement in terms of robustness to sub-pixel image translations, as shown in Section 4.

**First downsample layer** Unlike other CNN architectures that were examined in the context of aliasing prevention, ConvNeXt does not have a non-linearity before the first downsampling layer. Thus, due to the commutativity of convolutions with the LPFs, we cannot replace the first downsampling operation with BlurPool — since this is equivalent to applying a low-pass filter directly on the input, effectively reducing its resolution. Such composition may prevent the model from using high-frequency features and lead to a reduction in the model's accuracy. To solve this problem, we add an additional activation function before the first BlurPool. For computation efficiency, instead of using the regular scheme which requires upsampling be-

fore the activation, we replace the usual activation $\mathrm{Poly}_2(x)$ with

$$\mathrm{LPFPoly}_2(x) = a_0 + a_1 x + a_2 x \cdot \mathrm{LPF}_{\frac{3}{4}}(x) \ . \quad (5)$$

This modification of the polynomial activation leads to a smaller increase of the signal bandwidth. Thus, it does not require upsampling to avoid aliasing when it is followed by an LPF, as in the first BlurPool. Specifically, since it is followed by a BlurPool with a cutoff $1/4$, The maximally allowed cutoff for the LPF-Poly's filter is $3/4$. More details on this activation function can be found in Appendix F.2.

## 4. Experiments

We compare our Alias-Free Convnet (AFC) model to the baseline ConvNeXt model and to the previous integer shift-invariant method Adaptive Polyphase Sampling (APS) [4]. We implemented all models with cyclic convolutions and trained them on ImageNet [6] according to the ConvNeXt training regime [19]. The experiments were conducted with circular translations similarly to the setting in previous works [4, 33]. For sub-pixel translations, we used our "ideal upsampling" implementation (see Algorithm 2 in Appendix G); translation by $m/n$ pixels was conducted by upsampling by $n$, translating by $m$ pixels and downsampling by $n$.

### 4.1. Shift equivariance

Our model is designed to be not only shift-invariant (in terms of classification output), but also to have a Feature-Extractor that is shift-equivariant w.r.t. to the continuous domain. We verified this property by examining the response of the output of each of the layers to a translation of $\frac{1}{2}$ pixel in the input image. This was done by propagating the two translated inputs and measuring the difference between their outputs in each layer, after upsampling back to the input's spatial size. The results in Figure 3 show, in each layer, the normalized difference between the two translated layer outputs $y^0$ and $y^1$, after they were averaged across all $HW$ pixels (indexed by $i, j$) and $C$ channels (indexed by $c$),

$$\mathrm{diff} \triangleq \frac{1}{CHW} \sum_{c,i,j} \frac{\left|y^0_{c,i,j} - y^1_{c,i,j}\right|}{\max\left(\left|y^0_{c,i,j}\right|, \left|y^1_{c,i,j}\right|\right) + \varepsilon} \ , \quad (6)$$

where $\varepsilon = 10^{-9}$ was added in the denominator to avoid division by 0. The results show that ConvNeXt-AFC has only a negligible difference in the continuous representation of the translated responses at each layer, *e.g.* $y^0 = y^1$, which means it is indeed shift-equivariant w.r.t. the continuous domain (up to numerical error). In contrast, in the case of the baseline and APS models, the upsampled signals differ by more than 50% across all the layers.

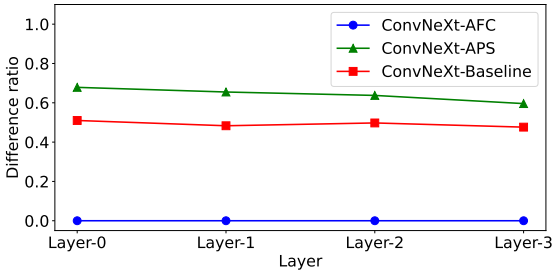


Figure 3. **Shift-equivariance measure w.r.t. continuous signal.** The averaged difference (Eq. (6)) for $1/2$ pixel translated inputs (y-axis), across all layers (x-axis). This experiment was run on 64 random samples from the validation set. While the AFC model has practically 0 difference, the baseline and APS models have at least 50% difference across all layers.

### 4.2. Consistency and Classification accuracy

The main measure used so far to quantify the shift-invariance of a model is called "consistency" [2,4], which is the percentage of predictions changed on the test set following an image shift. Previously this measure has been used with integer shift values, however, in Table 1 we test it also under sub-pixel shifts. We see that the changes we add to the baseline model gradually improve its consistency, until we reach 100%. In contrast, the previous APS approach [4] is near 100% consistent to integer shifts (due to numerical accuracy), but for fractional shifts, it only has slightly higher consistency than the baseline. Even though the alias-free modifications in our model lead to perfect consistency, they cause a 1.08% reduction in the (standard) test accuracy. We conclude that the main source of accuracy reduction is the modification of the normalization layer, as explained in Section 3. However, as we shall see next, despite such a reduction in accuracy, our model outperforms the previous models in adversarial shifts setting, due to its increased robustness.

### 4.3. Translation robustness

Since standard models are not invariant to image translation, this might be exploited as a very easy form of a "black-box" adversarial attack: we simply move the image until we notice the prediction is changed. We examine this vulnerability, to assess each model's actual robustness to translations. For each sample, we performed all possible translations in some set $T$, and checked the resulting classification for each shift. We define the adversarial accuracy corresponding to $T$ as the portion of samples that are classified correctly for all translations in $T$. We tested three types of basic translation grids — Integer, Half pixel and Fractional:

$$T_{\mathrm{integer}} = \{(i,\ j)\ |\ 1 \le i, j \le 31\} \quad (7)$$

$$T_{\mathrm{half}} = \left\{\left(\frac{i}{2},\ \frac{j}{2}\right)\ |\ 1 \le i, j \le 63\right\} \quad (8)$$

| Model modification | Test accuracy | Change | Integer shift consistency | Change | Fractional shift consistency | Change |
|---|---|---|---|---|---|---|
| ConvNeXt-Baseline [19] | 82.12 | | 94.816 | | 92.034 | |
| + Polynomial activation | 81.77 | -0.35 | 95.126 | 0.31 | 92.708 | 0.67 |
| + BlurPool | 78.99 | -3.12 | 96.635 | 1.82 | 96.572 | 4.54 |
| + First layer activation | 81.51 | -0.61 | 97.354 | 2.54 | 97.347 | 5.31 |
| + AF LayerNorm | 80.66 | -1.46 | 97.030 | 2.21 | 96.990 | 4.96 |
| + Activation upsample (ConvNeXt-AFC, ours) | 81.04 | -1.08 | 100.000 | 5.18 | 100.000 | 7.97 |
| ConvNeXt-APS [4] | 82.11 | -0.01 | 99.998 | 5.18 | 93.227 | 1.19 |

Table 1. **Alias-free modifications ImageNet accuracy and shift-consistency effect.** Integer shift consistency is defined as the percentage of test samples that did not change their prediction following a random integer translation. Fractional shift consistency is defined as the percentage of test samples that did not change their prediction following a random half-pixel translation. Consistency was averaged on five runs on ImageNet validation set with random seeds. The final AFC model is 100% consistent to both integer and fractional translations. Note that though the APS model [4] exhibits near 100% integer shifts consistency (as expected), it has only slightly better consistency than the baseline model in terms of fractional shift consistency.

| Model | Integer grid | Half-pixel grid | Fractional grid |
|---|---|---|---|
| ConvNeXt-Baseline [19] | 76.63 | 73.65 | 77.82 |
| ConvNeXt-APS [4] | 82.11 | 79.68 | 76.31 |
| ConvNeXt-AFC (ours) | 81.04 | 81.04 | 81.04 |

Table 2. **Translation adversarial accuracy (ImageNet).** Adversarial accuracy defined as the percentage of correctly classified samples for each translation in the corresponding set: Eq. (7), Eq. (8) or Eq. (9) with $k = 12$.

$$T_{\text{frac},k} = \left\{ \left( \frac{m_1}{n_1}, \frac{m_2}{n_2} \right) \mid 1 \leq m_{1,2} \leq n_{1,2} \leq k \right\} \quad (9)$$

In Table 2 we observe the adversarial robustness with respect to these translation sets. In the baseline model the test accuracy of 82.1% drops to 76.63% for integer grid and to 73.65% for half-pixel grid accuracy. This significant drop reflects that more than 10% of the correctly classified test set samples may be misclassified due to translations. The APS model [4] is, by construction, robust to integer translations and therefore has no accuracy reduction in the integer grid. However, it gets even worse results than the baseline in fractional adversarial accuracy (76.31% vs 77.82%). In contrast, our AFC model is invariant to any of these shifts, and therefore its accuracy remains constant at 81.04%, surpassing the other models. This robustness is 'certified', and will not be compromised with larger translation sets, or other types of attacks (*e.g.*, white box attacks) which can potentially decrease the performance of the other models even more. We repeat this experiment on "Out of distribution" data using ImageNet-C that contains common corruptions of the ImageNet images. The results in Appendix A show that the APS and Baseline models are even more vulnerable to fractional translation attacks on corrupted images, while AFC adversarial accuracy does not change.

### 4.4. Robustness to other shifts

We next test the models' robustness to other types of translations, where our model's shift-invariance guarantee conditions are not satisfied.

**Zero-padding, bilinear-interpolation** We tested the models' robustness to translation using the framework presented by Engstrom *et al*. [8], originally designed to test the robustness of classification models to translations and rotations. We zero-pad the images by 8 pixels and translate by (a possibly fractional) amount limited by 8 pixels, so there are no artifacts due to circular translations, nor data loss. The remaining parts are zero-padded and fractional translations are done using bilinear interpolation (see Fig. 9 in Appendix E). The results in Figure 4 (top) show the models' adversarial accuracy to this attack with different grid sizes. Although our model is not perfectly invariant to the performed translations due to the bilinear interpolation, it outperforms the other models by more than 4% at the largest tested grid.

**Crop-shift** In the experiments above, we used the common ImageNet input: the $224 \times 224$ center crop of the original $256 \times 256$ image. In contrast, in this experiment, we adversarially translated the cropped area, modeling translating a camera w.r.t. the scene (see Fig. 8 in Appendix E). We measure the adversarial accuracy of translations by up to $m$ integer pixels in each direction (*i.e.* grid search at size $(2m+1) \times (2m+1)$). The results in Figure 4 (bottom) show that our model is more robust to this kind of translation, which are not cyclic, include data loss, and are integer-valued. We additionally evaluate the original ConvNeXt model (zero-pad convolutions) which interestingly has the worst robustness in this setting.

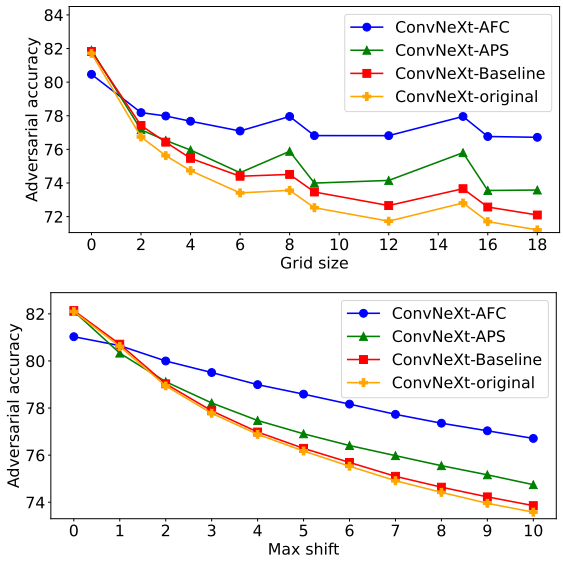


Figure 4. **Adversarial accuracy for other types of shifts. Top**: Zero-padding, bilinear interpolation results. AFC is the most robust model for all tested grid sizes. **Bottom**: Crop-shift results. AFC is the most robust model for $m \geq 2$. The accuracy improvement over the baseline and APS models reaches to $2.9\%$ and $2\%$ respectively for the strongest attack in our scope ($m = 10$).

## 5. Related work

Modern Convolutional Neural Networks use downsampling operations such as pooling and strided convolutions to increase the net's receptive field, with lower computation cost than using larger kernels for that matter. It was shown that this architectural design breaks the shift-equivariance property of the convolution operation due to the aliasing effect [2], leading CNNs to be not shift-invariant. Even though it was shown this property can be partially learned using appropriate data augmentation [12], other works tried to architecturally regain shift-invariance.

Another work [33] suggested shift-equivariance could be maintained by reducing aliasing using low-pass filters (LPFs) before downsampling [22]. Others [34] improved the LPF method by using content-aware adaptive filters. These changes were shown to improve convnets robustness to translations, as well as accuracy and generalization, yet another work showed that focusing on circular shifts may induce adversarial attack vulnerabilities [27].

Instead of tackling the aliasing problem, other works suggested solving shift variance by using adaptive subsampling grids [4, 24, 31]. This approach was shown ability to produce perfect shift-invariance in image classification tasks. Yet, as it does not eliminate the aliasing effects, it does not produce shift-invariance to fractional shifts, and it does not ensure shift-equivariance of the internal representations.

Since it is known that aliasing in discrete signals is caused by non-linearities in addition to subsampling, a few studies suggested methods for alias-free activation functions. Karras *et al*. [17] suggested using upsampling before non-linearities to reduce aliasing in generative models, which cause failure in embedding "high-frequency features" such as textures in their outputs. The idea of using polynomial non-linearities to battle aliasing has been mentioned previously [7, 22]. Franzen and Wand [9] have recently shown this methodology can be used to improve rotation-equivariance. However, it has never been applied in a complete alias-free setting, nor in modern-scale deep networks. Other smooth activation functions have been suggested as well [16, 29], yet they do not completely eliminate aliasing.

It is worth mentioning that other equivariance properties have been studied as well, such as rotation, reflection and group equivariance [3, 5, 20, 21, 25, 30–32]. This work is focused on the specific property of shift-invariance in CNNs for image classification.

## 6. Discussion

In this paper we proposed the Alias-Free Convnet, which for the first time, is guaranteed to eliminate any aliasing effects in the model, to ensure the output is invariant to any input shifts (even sub-pixel ones), and to ensure the internal representations are equivariant to any shifts (even sub-pixel ones). We demonstrate this numerically and show this leads to (certified) high performance under adversarial shift-based attacks — in contrast to existing models which degrade in performance. However, this comes at a cost, such as a 1.08% reduction in standard test accuracy (as methods that increase robustness often reduce accuracy) and increased computation cost. We further discuss these limitations as well as limitations regarding the certified robustness conditions in Appendix B. We discuss possible future work and applications of the AFC in Appendix C, and the potential of polynomial activation functions in general (*i.e*., beyond AFC) in Appendix D.

**Acknowledgments** The research of DS was funded by the European Union (ERC, A-B-C-Deep, 101039436). Views and opinions expressed are however those of the author only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency (ERCEA). Neither the European Union nor the granting authority can be held responsible for them. DS also acknowledges the support of Schmidt Career Advancement Chair in AI. TM was supported by grant 2318/22 from the Israel Science Foundation and by the Ollendorff Center, ECE faculty, Technion.

## References


[1] Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. 2 2017. 16

[2] Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *Journal of Machine Learning Research*, 20:1–25, 2019. 1, 6, 8, 11

[3] Michael M. Bronstein, Joan Bruna, Yann Lecun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 11 2016. 8

[4] Anadi Chaman and Ivan Dokmanić. Truly shift-invariant convolutional neural networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3772–3782, 11 2020. 1, 4, 6, 7, 8, 12, 20

[5] Valentin Delchevalerie, Adrien Bibal, Benoît Frénay, and Alexandre Mayer. Achieving Rotational Invariance with Bessel-Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 34:28772–28783, 12 2021. 8

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 6 2009. 6, 15

[7] S. Emmy Wei. Aliasing-Free Nonlinear Signal Processing Using Implicitly Defined Functions. *IEEE Access*, 10:76281–76295, 2022. 8

[8] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the Landscape of Spatial Robustness. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:3218–3238, 12 2017. 1, 7, 15

[9] Daniel Franzen and Michael Wand. General Nonlinearities in SO(2)-Equivariant CNNs. *Advances in Neural Information Processing Systems*, 34:9086–9098, 12 2021. 8

[10] Joseph W. Goodman and Mary E. Cox. Introduction to Fourier Optics. *Physics Today*, 22(4):97–101, 4 1969. 12

[11] Vikas Gottemukkula. POLYNOMIAL ACTIVATION FUNCTIONS. Technical report, 2019. 2, 4, 13, 14

[12] Suriya Gunasekar. Generalization to translation shifts: a study in architectures and augmentations, 7 2022. 8

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December:770–778, 12 2015. 14

[14] Dan Hendrycks and Thomas Dietterich. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *7th International Conference on Learning Representations, ICLR 2019*, 3 2019. 11

[15] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1 1989. 2

[16] Md Tahmid Hossain, Shyh Wei Teng, Ferdous Sohel, and Guojun Lu. Anti-aliasing Deep Image Classifiers using Novel Depth Adaptive Blurring and Activation Function, 10 2021. 8

[17] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-Free Generative Adversarial Networks, 6 2021. 1, 2, 3, 8, 12, 13, 17

[18] Patrick Kidger and Terry Lyons. Universal Approximation with Deep Narrow Networks. *Proceedings of Machine Learning Research*, TBD:1–22, 2020. 2

[19] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, Saining Xie, and Facebook AI Research. A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 4, 6, 7, 12

[20] Marco Manfredi and Yu Wang. Shift Equivariance in Object Detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12540 LNCS:32–45, 8 2020. 8

[21] Mingqiang Ning, Jinsong Tang, Heping Zhong, Haoran Wu, Peng Zhang, and Zhisheng Zhang. Scale-Aware Network with Scale Equivariance. *Photonics 2022, Vol. 9, Page 142*, 9(3):142, 2 2022. 8

[22] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Prentice-hall Englewood Cliffs, second edition, 1999. 8

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 4

[24] Renan A. Rojas-Gomez, Teck-Yian Lim, Alexander G. Schwing, Minh N. Do, and Raymond A. Yeh. Learnable Polyphase Sampling for Shift Invariant and Equivariant Convolutional Networks. 10 2022. 8

[25] David W. Romero, Erik J. Bekkers, Jakub M. Tomczak, and Mark Hoogendoorn. Attentive Group Equivariant Convolutional Networks. 2 2020. 8

[26] Michael Schoberl, Wolfgang Schnurrer, Alexander Oberdorster, Siegfried Fossel, and Andre Kaup. Dimensioning of optical birefringent anti-alias filters for digital cameras. In *2010 IEEE International Conference on Image Processing*, pages 4305–4308. IEEE, 9 2010. 12

[27] Vasu Singla, Songwei Ge, Ronen Basri, and David Jacobs. Shift Invariance Can Reduce Adversarial Robustness, 3 2021. 8

[28] J. Staal, M.D. Abramoff, M. Niemeijer, M.A. Viergever, and B. van Ginneken. Ridge-Based Vessel Segmentation in Color Images of the Retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 4 2004. 16

[29] Cristina Vasconcelos, Hugo Larochelle, Vincent Dumoulin, Nicolas Le Roux, and Ross Goroshin. An Effective Anti-Aliasing Approach for Residual Networks. 11 2020. 8

[30] Maurice Weiler and Gabriele Cesa. General $E(2)$-Equivariant Steerable CNNs. *Advances in Neural Information Processing Systems*, 32, 11 2019. 8

[31] Jin Xu, Hyunjik Kim, Tom Rainforth, and Yee Whye Teh. Group Equivariant Subsampling. *Advances in Neural Information Processing Systems*, 8:5934–5946, 6 2021. 8

[32] Raymond A Yeh, Yuan-Ting Hu, Mark Hasegawa-Johnson, and Alexander G Schwing. Equivariance Discovery by Learned Parameter-Sharing. 2022. 8

[33] Richard Zhang. Making convolutional networks shift-invariant again. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:12712–12722, 2019. 1, 4, 6, 8, 20

[34] Xueyan Zou, Fanyi Xiao, Zhiding Yu, and Yong Jae Lee. Delving Deeper into Anti-aliasing in ConvNets. *International Journal of Computer Vision 2022*, pages 1–15, 8 2020. 8