

# “Fourier Series Expansion Based Filter Parametrization for Equivariant Convolutions”: Supplementary Material

Qi Xie, Qian Zhao, Zongben Xu and Deyu Meng

**Abstract**—This supplementary material presents the proofs for Remark 1, Remark 2 and Theorem 1 in the main paper. More implementation details of the proposed method and more experimental results are also provided for better reference of readers. The code of our method is available at <https://github.com/XieQi2015/F-Conv>.

## 1 PROOF TO THEORETICAL RESULTS

### 1.1 Proof of Remark 1

**Notations.** Firstly, let us review the notations defined in the main text. We denote the 2D Fourier bases as following:

$$\begin{aligned}\phi_{kl}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}[k, l] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right), \\ \phi_{kl}^s(x) &= \Omega(x) \sin\left(\frac{2\pi}{ph}[k, l] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right),\end{aligned}\quad (1)$$

where  $k, l = 0, 1, \dots, p-1$  and  $\Omega(x) \geq 0$  is a radial mask function that satisfies  $\Omega(x) = 0$  if  $\|x\| > (p+1/2)/h$  (please refer to Section 4 and Eq. (76) for more details of  $\Omega(x)$ ). We denote the proposed bases in the following formulation:

$$\begin{aligned}\varphi_{kl}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left[k - \left\lfloor \frac{p}{2} \right\rfloor, l - \left\lfloor \frac{p}{2} \right\rfloor\right] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right), \\ \varphi_{kl}^s(x) &= \Omega(x) \sin\left(\frac{2\pi}{ph}\left[k - \left\lfloor \frac{p}{2} \right\rfloor, l - \left\lfloor \frac{p}{2} \right\rfloor\right] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right),\end{aligned}\quad (2)$$

where  $k, l = 0, 1, \dots, p-1$ , and  $\Omega(x)$  is the aforementioned radial mask function. Then we can prove following conclusion:

**Remark 1.** For any mesh size  $h \in \mathbb{R}$ , filter size  $p \in \mathbb{N}_+$ , and grid point  $x$  on the  $p \times p$  mesh of  $[(1-p)h/2, (p-1)h/2]^2$ , i.e.,  $x_1 = (i - (p-1)/2)h$ ,  $x_2 = (j - (p-1)/2)h$ ,  $\forall i, j = 0, 1, \dots, p-1$ , let  $k, l = 0, 1, \dots, p-1$ , and then,

$$\begin{aligned}\phi_{kl}^c(x) &= s(k, l) \cdot \varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x), \\ \phi_{kl}^s(x) &= s(k, l) \cdot \varphi_{\mathcal{I}(k), \mathcal{I}(l)}^s(x),\end{aligned}\quad (3)$$

where  $\varphi_{kl}^c$  and  $\varphi_{kl}^s$  are defined in (2),  $\phi_{kl}^c$ ,  $\phi_{kl}^s$  are defined in (1),  $\mathcal{I}(\cdot) = ((\cdot) + \lfloor p/2 \rfloor) \% p$ , and  $s(k, l) \in \{-1, 1\}$ , satisfying  $s(k, l) = \text{sign}(k - p/2 + \epsilon)^{p-1} \cdot \text{sign}(l - p/2 + \epsilon)^{p-1}$ ,  $0 < \epsilon < 1/2$ .

*Proof.* 1) When  $p$  is an odd number, then we have

$$\begin{aligned}x_1 &= \left(i - \frac{(p-1)}{2}\right)h = mh, \quad \forall m, n \in \mathbb{N} \\ x_2 &= \left(j - \frac{(p-1)}{2}\right)h = nh, \quad \forall m, n \in \mathbb{N}.\end{aligned}\quad (4)$$

Qi Xie, Qian Zhao, Zongben Xu and Deyu Meng (corresponding author) are with School of Mathematics and Statistics and Ministry of Education Key Lab of Intelligent Networks and Network Security, Xi'an Jiaotong University, Shaanxi, P.R.China.

Email: {xie.qi, timmy.zhaoqian, zbxu, dymeng}@mail.xjtu.edu.cn

In this case,  $s(k, l) = 1$  and we can obtain following deductions.

a) If  $k < p/2$  and  $l < p/2$ , we can obtain  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor$ , and  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor$ . Then, we have

$$\begin{aligned}\varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2)\right) \\ &= \phi_{k,l}^c(x),\end{aligned}\quad (5)$$

which means that Eq. (3) is satisfied. In the similar way, we can prove  $\varphi_{kl}^s(x) = \phi_{kl}^s(x)$ , which then follows the results of Eq. (3).

b) If  $k < p/2$  and  $l \geq p/2$ , then  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor - p$ , and  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor - p$ . We can then obtain

$$\begin{aligned}\varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + (l-p)x_2)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2) - \frac{2\pi}{h} \cdot x_2\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2) - 2n\pi\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2)\right) \\ &= \phi_{kl}^c(x).\end{aligned}\quad (6)$$

In the similar way, we can also deduce that  $\varphi_{kl}^s(x) = \phi_{kl}^s(x)$ , which follows the results of Eq. (3) in this case.

c) If  $k \geq p/2$  and  $l < p/2$ , the result of Eq. (3) can be proved in the similar way as (b).

d) If  $k \geq p/2$  and  $l \geq p/2$ , then  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor - p$ , and  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor - p$ . We can then have

$$\begin{aligned}\varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}((k-p)x_1 + (l-p)x_2)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2) - \frac{2\pi}{h}(x_1 + x_2)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}(kx_1 + lx_2) - 2(m+n)\pi\right)\end{aligned}\quad (7)$$

$$= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) \right)$$

$$= \phi_{kl}^c(x).$$

In the similar way, we can deduce  $\varphi_{kl}^s(x) = \phi_{kl}^s(x)$ , which follows the results of Eq. (3) in this case.

2) When  $p$  is an even number, then

$$x_1 = \left( i - \frac{(p-1)}{2} \right) h = \left( m + \frac{1}{2} \right) h, \quad \forall m, n \in \mathbb{N}$$

$$(8)$$

$$x_2 = \left( j - \frac{(p-1)}{2} \right) h = \left( m + \frac{1}{2} \right) h, \quad \forall m, n \in \mathbb{N}.$$

a) If  $k \geq p/2$  and  $l < p/2$ ,  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor$ ,  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor$ ,  $s(k, l) = 1$ . Similar as the deduction in (5), we can prove that (3) is satisfied in this case.

b) If  $k \geq p/2$  and  $l > p/2$ , then  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor$ ,  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor - p$ , and  $s(k, l) = -1$ . We can deduce that

$$\begin{aligned} \varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x) &= \Omega(x) \cos \left( \frac{2\pi}{ph} (x_1 + (l-p)x_2) \right) \\ &= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) - \frac{2\pi}{h} x_2 \right) \\ &= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) - (2n+1)\pi \right) \\ &= -\Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) \right) \\ &= -\phi_{kl}^c(x). \end{aligned} \quad (9)$$

In the similar way, we can deduce that  $\varphi_{kl}^s(x) = -\phi_{kl}^s(x)$ , which follows the results of Eq. (3) in this case..

c) If  $k \geq p/2$  and  $l < p/2$ , then the result can be proved in the similar way as (b).

d) If  $k \geq p/2$  and  $l \geq p/2$ , then  $\mathcal{I}(k) = k + \lfloor p/2 \rfloor - p$ ,  $\mathcal{I}(l) = l + \lfloor p/2 \rfloor - p$  and  $s(k, l) = 1$ . We can then obtain

$$\begin{aligned} \varphi_{\mathcal{I}(k), \mathcal{I}(l)}^c(x) &= \Omega(x) \cos \left( \frac{2\pi}{ph} ((k-p)x_1 + (l-p)x_2) \right) \\ &= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) - \frac{2\pi}{h} (x_1 + x_2) \right) \\ &= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) - 2(m+n+1)\pi \right) \\ &= \Omega(x) \cos \left( \frac{2\pi}{ph} (kx_1 + lx_2) \right) \\ &= \phi_{kl}^c(x). \end{aligned} \quad (10)$$

Similarly, we can also deduce that  $\varphi_{kl}^s(x) = \phi_{kl}^s(x)$ , which follows the results of Eq. (3) in this case.  $\square$

## 1.2 Proof of Remark 2

**Notations.** For an input  $r \in C^\infty(\mathbb{R}^2)$  and a transformation  $\tilde{A} \in O(2)$ ,  $\tilde{A}$  acts on  $r$  by

$$\pi_{\tilde{A}}^R[r](x) = r(\tilde{A}^{-1}x), \quad \forall x \in \mathbb{R}^2. \quad (11)$$

For a feature map  $e \in C^\infty(E(2))$  and a transformation  $\tilde{A} \in O(2)$ ,  $\tilde{A}$  act on  $e$  by

$$\pi_{\tilde{A}}^E[e](x, A) = e(\tilde{A}^{-1}x, \tilde{A}^{-1}A), \quad \forall (x, A) \in E(2). \quad (12)$$

$\Psi$  denotes the convolution on the input layer, which maps an input  $r \in C^\infty(\mathbb{R}^2)$  to a feature map defined on  $E(2)$ :

$$\Psi[r](y, A) = \int_{\mathbb{R}^2} \varphi_{in}(A^{-1}x) r(y-x) d\sigma(x), \quad (13)$$

$\forall (y, A) \in E(2)$ , where  $\sigma$  is a measure on  $\mathbb{R}^2$  and  $\varphi$  is the proposed parameterized filter.  $\Phi$  denotes the convolution on the intermediate layer, which maps a feature map  $e \in C^\infty(E(2))$  to another feature map defined on  $E(2)$ :

$$\Phi[e](y, B) = \int_{O(2)} \int_{\mathbb{R}^2} \varphi_A(B^{-1}x) e(y-x, BA) d\sigma(x) dv(A), \quad (14)$$

$\forall (y, B) \in E(2)$ , where  $v$  is a measure on  $O(2)$ ,  $A, B \in O(2)$  denotes orthogonal transformations in the considered group, and  $\varphi_{\tilde{A}}$  indicates the filter with respect to the channel of feature map indexed by  $\tilde{A}$ , i.e.,  $e(x, A)|_{A=\tilde{A}}$ .  $\Upsilon$  denotes the convolution on final layer, which maps a feature map  $e \in C^\infty(E(2))$  to a function defined on  $\mathbb{R}^2$ :

$$\Upsilon[e](y) = \int_{O(2)} \int_{\mathbb{R}^2} \varphi_{out}(B^{-1}x) e(y-x, B) d\sigma(x) dv(B), \quad (15)$$

$\forall y \in \mathbb{R}^2$ . Then we can prove the following result:

**Remark 2.** For  $r \in C^\infty(\mathbb{R}^2)$ ,  $e \in C^\infty(E(2))$  and  $\tilde{A} \in O(2)$ , the following results are satisfied:

$$\begin{aligned} \Psi \left[ \pi_{\tilde{A}}^R[r] \right] &= \pi_{\tilde{A}}^E [\Psi[r]], \\ \Phi \left[ \pi_{\tilde{A}}^E[e] \right] &= \pi_{\tilde{A}}^E [\Phi[e]], \\ \Upsilon \left[ \pi_{\tilde{A}}^E[e] \right] &= \pi_{\tilde{A}}^R [\Upsilon[e]], \end{aligned} \quad (16)$$

where  $\pi_{\tilde{A}}^R$ ,  $\pi_{\tilde{A}}^E$ ,  $\Psi$ ,  $\Phi$  and  $\Upsilon$  are defined by (11), (12), (13), (14) and (15), respectively.

*Proof.* (1) For any  $y \in \mathbb{R}^2$  and  $A \in O(2)$ , we can obtain

$$\begin{aligned} &\Psi \left[ \pi_{\tilde{A}}^R[r] \right] (y, A) \\ &= \int_{\mathbb{R}^2} \varphi_{in}(A^{-1}x) \pi_{\tilde{A}}^R[r](y-x) d\sigma(x) \\ &= \int_{\mathbb{R}^2} \varphi_{in}(A^{-1}x) r(\tilde{A}^{-1}(y-x)) d\sigma(x). \end{aligned} \quad (17)$$

Let  $\hat{x} = \tilde{A}^{-1}x$ , since  $|\det(\tilde{A})| = 1$ , and we have

$$\begin{aligned} &\int_{\mathbb{R}^2} \varphi_{in}(A^{-1}x) r(\tilde{A}^{-1}(y-x)) d\sigma(x), \\ &= \int_{\mathbb{R}^2} \varphi_{in} \left( \left( \tilde{A}^{-1}A \right)^{-1} \hat{x} \right) r(\tilde{A}^{-1}y - \hat{x}) d\sigma(\hat{x}) \\ &= \Psi[r](\tilde{A}^{-1}y, \tilde{A}^{-1}A) \\ &= \pi_{\tilde{A}}^E [\Psi[r]](y, A). \end{aligned} \quad (18)$$

This proves that  $\Psi \left[ \pi_{\tilde{A}}^R[r] \right] = \pi_{\tilde{A}}^E [\Psi[r]]$ .

(2) Similar to the proof in (1), for any  $y \in \mathbb{R}^2$  and  $B \in O(2)$ , we can obtain

$$\begin{aligned} & \Phi \left[ \pi_{\tilde{A}}^E [e] \right] (y, B) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_A (B^{-1}x) \pi_{\tilde{A}}^E [e] (y - x, BA) d\sigma(x) v(A) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_A (B^{-1}x) \\ &\quad e(\tilde{A}^{-1}(y - x), \tilde{A}^{-1}BA) d\sigma(x) v(A) \quad (19) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_A \left( (\tilde{A}^{-1}B)^{-1} \hat{x} \right) \\ &\quad e(\tilde{A}^{-1}y - \hat{x}, \tilde{A}^{-1}BA) d\sigma(x) v(A) \\ &= \Phi [e] (\tilde{A}^{-1}y, \tilde{A}^{-1}B) \\ &= \pi_{\tilde{A}}^E [\Phi [e]] (y, B). \end{aligned}$$

(3) For any  $y \in \mathbb{R}^2$ , we can deduce that

$$\begin{aligned} & \Upsilon \left[ \pi_{\tilde{A}}^E [e] \right] (y) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_{out} (B^{-1}x) \pi_{\tilde{A}}^E [e] (y - x, B) d\sigma(x) v(B) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_{out} (B^{-1}x) \\ &\quad e(\tilde{A}^{-1}(y - x), \tilde{A}^{-1}B) d\sigma(x) v(B) \quad (20) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_{out} \left( (\tilde{A}^{-1}B)^{-1} \hat{x} \right) \\ &\quad e(\tilde{A}^{-1}y - \hat{x}, \tilde{A}^{-1}B) d\sigma(x) v(B). \end{aligned}$$

Let  $C = \tilde{A}^{-1}B$ , and then we have

$$\begin{aligned} & \int_{\mathbb{R}^2} \int_{O(2)} \varphi_{out} \left( (\tilde{A}^{-1}B)^{-1} \hat{x} \right) \\ &\quad e(\tilde{A}^{-1}y - \hat{x}, \tilde{A}^{-1}B) d\sigma(x) v(B) \\ &= \int_{\mathbb{R}^2} \int_{O(2)} \varphi_{out} (C^{-1}\hat{x}) \\ &\quad e(\tilde{A}^{-1}y - \hat{x}, C) d\sigma(x) v(C) \quad (21) \\ &= \Upsilon [e] (\tilde{A}^{-1}y) \\ &= \pi_{\tilde{A}}^R [\Upsilon [e]] (y). \end{aligned}$$

This proves that  $\Upsilon \left[ \pi_{\tilde{A}}^E [r] \right] = \pi_{\tilde{A}}^R [\Upsilon [r]]$ .  $\square$

### 1.3 Proof of Theorem 1

**Notations.** We assume that an image  $I \in R^{n \times n}$  represents a two-dimensional grid function obtained by discretizing a smooth function, i.e., for  $i, j = 1, 2, \dots, n$ ,

$$I_{ij} = r(x_{ij}), \quad (22)$$

where  $x_{ij} = ((i - \frac{n+1}{2})h, (j - \frac{n+1}{2})h)^T$ . We represent  $F$  as a three-dimensional grid function sampled from a smooth function  $e : \mathbb{R}^2 \times S \rightarrow \mathbb{R}$ , i.e., for  $i, j = 1, 2, \dots, n$ ,

$$F_{ij}^A = e(x_{ij}, A), \quad (23)$$

where  $x_{ij} = ((i - \frac{n+1}{2})h, (j - \frac{n+1}{2})h)^T$  and  $A \in S$ ,  $S$  is a subgroup of  $O(2)$ . For  $i, j = 1, 2, \dots, p$ , and  $A, B \in S$ , we have

$$\begin{aligned} \tilde{\Psi}_{ij}^A &= \varphi_{in} (A^{-1}x_{ij}), \\ \tilde{\Phi}_{ij}^{B,A} &= \varphi_A (B^{-1}x_{ij}), \\ \tilde{\Upsilon}_{ij}^A &= \varphi_{out} (A^{-1}x_{ij}), \end{aligned} \quad (24)$$

where  $x_{ij} = ((i - (p+1)/2)h, (j - (p+1)/2)h)^T$ , and  $\varphi$  and  $\varphi_A$  are parameterized filters. Let

$$\begin{aligned} x_{ij} &= \left( \left( i - \frac{p+1}{2} \right) h, \left( j - \frac{p+1}{2} \right) h \right)^T, \\ y_{ij} &= \left( \left( i - \frac{n+p+2}{2} \right) h, \left( j - \frac{n+p+2}{2} \right) h \right)^T. \end{aligned} \quad (25)$$

For  $\forall A \in S$  and  $i, j = 1, 2, \dots, n$ , the convolution of  $\tilde{\Psi}$  and  $I$  is

$$\left( \tilde{\Psi} * I \right)_{ij}^A = \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{in} \left( A^{-1}x_{\tilde{i}\tilde{j}} \right) r \left( y_{ij} - x_{\tilde{i}, \tilde{j}} \right), \quad (26)$$

where  $\Lambda$  is a set of indexes, denoted as  $\Lambda = \{(i, j) | i, j = 1, 2, \dots, p\}$ , and  $*$  represents the common 2D convolution. For any  $B \in S$  and  $i, j = 1, 2, \dots, n$ , the convolution of  $\tilde{\Phi}$  and  $F$  is

$$\left( \tilde{\Phi} * F \right)_{ij}^B = \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, A \in S} \varphi_A \left( B^{-1}x_{\tilde{i}\tilde{j}} \right) e \left( y_{ij} - x_{\tilde{i}, \tilde{j}}, BA \right), \quad (27)$$

where  $\Lambda = \{(i, j) | i, j = 1, 2, \dots, p\}$ . For  $i, j = 1, 2, \dots, n$ , the convolution of  $\tilde{\Upsilon}$  and  $F$  is

$$\left( \tilde{\Upsilon} * F \right)_{ij} = \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, B \in S} \varphi_{out} \left( B^{-1}x_{\tilde{i}\tilde{j}} \right) e \left( y_{ij} - x_{\tilde{i}, \tilde{j}}, B \right) \quad (28)$$

where  $\Lambda = \{(i, j) | i, j = 1, 2, \dots, p\}$ .

The transformations on  $I$  and  $F$  are defined by

$$\begin{aligned} \left( \tilde{\pi}_{\tilde{A}}^R (I) \right)_{ij} &= \pi_{\tilde{A}}^R [r](x_{ij}), \quad \left( \tilde{\pi}_{\tilde{A}}^{\tilde{E}} (F) \right)_{ij}^A = \pi_{\tilde{A}}^E [e](x_{ij}, A), \\ \forall i, j &= 1, 2, \dots, n, \forall A, \tilde{A} \in S. \end{aligned} \quad (29)$$

Then we introduce following necessary lemmas.

**Lemma 1.** For smooth functions  $r : \mathbb{R}^2 \rightarrow \mathbb{R}$  and  $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}$ , let  $f(x) = \varphi (\tilde{A}^{-1}x) r(y - x)$ . If  $\forall x \in \mathbb{R}^2$ , the follow results are satisfied:

$$\begin{aligned} & |r(x)| \leq F_1, |\varphi(x)| \leq F_2, \\ & \|\nabla r(x)\| \leq G_1, \|\nabla \varphi(x)\| \leq G_2, \\ & \|\nabla^2 r(x)\| \leq H_1, \|\nabla^2 \varphi(x)\| \leq H_2, \\ & \forall \|x\| \geq (p+1/2)h, \varphi(x) = 0, \end{aligned} \quad (30)$$

where  $p, h > 0$ ,  $\nabla$  and  $\nabla^2$  denote the operators of gradient and Hessian matrices, respectively. Then,  $\forall \tilde{A} \in S, y \in \mathbb{R}$  the following results are satisfied:

$$\begin{aligned} & \max \left\{ \left| \frac{\partial^2 f(x)}{\partial x_1^2} \right|, \left| \frac{\partial^2 f(x)}{\partial x_2^2} \right| \right\} \leq C, \\ & \max \left\{ \left| \frac{d^2}{dx_2^2} \int_{\mathbb{R}} f(x) dx_1 \right|, \left| \frac{d^2}{dx_1^2} \int_{\mathbb{R}} f(x) dx_2 \right| \right\} \leq C(p+1)h, \end{aligned} \quad (31)$$

where  $C = F_1 H_2 + F_2 H_1 + 2G_1 G_2$ .

*Proof.* It is easy to deduce that

$$\begin{aligned} \left| \frac{\partial r(\tilde{A}y - x)}{\partial x_1} \right| &\leq \|\nabla r(x)\| \leq G_1, \\ \left| \frac{\partial \varphi(\tilde{A}x)}{\partial x_1} \right| &\leq \|\nabla \varphi(x)\| \leq G_2 \\ \left| \frac{\partial^2 r(\tilde{A}y - x)}{\partial x_1^2} \right| &\leq \|\nabla^2 r(x)\| \leq H_1, \\ \left| \frac{\partial^2 \varphi(\tilde{A}x)}{\partial x_1^2} \right| &\leq \|\nabla^2 \varphi(x)\| \leq H_2. \end{aligned} \quad (32)$$

Then we can obtain that

$$\begin{aligned} \left| \frac{\partial^2 f(x)}{\partial x_1^2} \right| &= \left| \frac{\partial (\varphi(\tilde{A}x) \partial r(y-x) + r(y-x) \partial \varphi(\tilde{A}x))}{\partial x_1^2} \right| \\ &= \left| \frac{\varphi(\tilde{A}x) \partial^2 r(y-x) + r(y-x) \partial^2 \varphi(\tilde{A}x) + 2\partial \varphi(\tilde{A}x) \partial r(y-x)}{\partial x_1^2} \right| \\ &\leq F_1 H_2 + F_2 H_1 + 2G_1 G_2. \end{aligned} \quad (33)$$

Since the same result can also be deduced for  $x_2$ , the proof of the first inequality is complete.

Besides,  $\forall \|x\| \geq (p+1/2)h$ ,  $\varphi(x) = 0$ , and only in the circular area with radius  $(p+1/2)h$ . We then have  $\left| \frac{\partial^2 f(x)}{\partial x_1^2} \right| > 0$ , and thus

$$\begin{aligned} &\left| \frac{d^2}{dx_1^2} \int_{\mathbb{R}} f(x) dx_1 \right| = \left| \int_{\mathbb{R}} \frac{\partial^2}{\partial x_1^2} f(x) dx_1 \right| \\ &\leq \int_{-\frac{p+1}{2}h}^{\frac{p+1}{2}h} \left| \frac{\partial^2}{\partial x_1^2} f(x) \right| dx_1 \\ &\leq C(p+1)h. \end{aligned} \quad (34)$$

Since the same result can be deduced for  $x_2$ , this completes the proof of the first Inequality.  $\square$

**Lemma 2.** For any  $f : \mathbb{R} \rightarrow \mathbb{R}$ , denote  $l$  as the linear interpolation function between points  $x = a$  and  $x = b$ , i.e.,

$$l(x) = \frac{(x-a)f(b) + (b-x)f(a)}{b-a}. \quad (35)$$

If  $f''$  is continuous in  $[a, b]$  and  $|f''(x)| \leq C$ , then it holds that

$$|f(x) - l(x)| \leq \frac{C}{8}(a-b)^2. \quad (36)$$

*Proof.* Based on the Taylor's theorem with Lagrange remainder, we can obtain

$$\begin{aligned} f(a) &= f(x) + (a-x)f'(x) + \frac{1}{2}(a-x)^2 f''(\xi) \\ f(b) &= f(x) + (b-x)f'(x) + \frac{1}{2}(b-x)^2 f''(\zeta). \end{aligned} \quad (37)$$

Besides, it is easy to deduce that

$$f(x) = \frac{(x-a)f(x) + (b-x)f(x)}{b-a}. \quad (38)$$

Thus,

$$\begin{aligned} &l(x) - f(x) \\ &= \frac{(x-a)(f(b) - f(x)) + (b-x)(f(a) - f(x))}{b-a} \\ &= (x-a) \left( \frac{1}{2}(b-x)^2 f''(\zeta) \right) + \\ &\quad (b-x) \left( \frac{1}{2}(a-x)^2 f''(\xi) \right) \\ &= \frac{(x-a)(b-x)}{2} \left( \frac{b-x}{b-a} f''(\zeta) + \frac{x-a}{b-a} f''(\xi) \right). \end{aligned} \quad (39)$$

Since  $|(x-a)(b-x)| \leq (b-a)^2/4$ , we can obtain

$$\begin{aligned} &|l(x) - f(x)| \\ &= \left| \frac{(x-a)(b-x)}{2} \left( \frac{b-x}{b-a} f''(\zeta) + \frac{x-a}{b-a} f''(\xi) \right) \right| \\ &\leq \left| \frac{(b-a)^2}{2} \left( \frac{b-x}{b-a} C + \frac{x-a}{b-a} C \right) \right| \\ &= \frac{C}{8}(b-a)^2. \end{aligned} \quad (40)$$

$\square$

**Lemma 3.** For smooth functions  $r : \mathbb{R}^2 \rightarrow \mathbb{R}$  and  $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}$ , if for  $x \in \mathbb{R}^2$ , the follow conditions are satisfied:

$$\begin{aligned} &|r(x)| \leq F_1, |\varphi(x)| \leq F_2, \\ &\|\nabla r(x)\| \leq G_1, \|\nabla \varphi(x)\| \leq G_2, \\ &\|\nabla^2 r(x)\| \leq H_1, \|\nabla^2 \varphi(x)\| \leq H_2, \\ &\forall \|x\| \geq (p+1/2)h, \varphi(x) = 0 \end{aligned} \quad (41)$$

where  $p, h > 0$ ,  $\nabla$  and  $\nabla^2$  denotes the operators of gradient and Hessian matrix, respectively, then,  $\forall \tilde{A} \in S, y \in \mathbb{R}$  the following results are satisfied:

$$\begin{aligned} &\left| \int_{R^2} \varphi(\tilde{A}^{-1}x) r(y-x) d\sigma(x) - \right. \\ &\quad \left. \sum_{i,j \in \Lambda} \varphi(\tilde{A}^{-1}x_{ij}) r(y-x_{ij}) h^2 \right| \leq \frac{(p+1)^2 C}{4} h^4, \end{aligned} \quad (42)$$

where  $A \in S$  and  $\Lambda = \{(i,j) | i, j = 1, 2, \dots, p\}$ ,  $x_{ij} = ((i-(p+1)/2)h, (j-(p+1)/2)h)^T$  and  $C = F_1 H_2 + F_2 H_1 + 2G_1 G_2$ .

*Proof.* 1) Denote  $f(x) = \varphi(\tilde{A}^{-1}x) r(y-x)$ , and we first prove that for any  $n \in \mathbb{Z}$ , it holds that

$$\left| \int_{\mathbb{R}} f\left(\begin{bmatrix} x_1 \\ nh \end{bmatrix}\right) dx_1 - \sum_{m \in \mathbb{Z}} f\left(\begin{bmatrix} mh \\ nh \end{bmatrix}\right) h \right| \leq \frac{C(p+1)}{8} h^3. \quad (43)$$

By **Lemma 1**, we can obtain that  $\left| \frac{\partial^2 f(x)}{\partial x_1^2} \right| \leq C$ . Then by **Lemma 2**, we can obtain that  $\forall m, n \in \mathbb{Z}, mh \leq x_1 < (m+1)h$ , it holds that,

$$\left| f\left(\begin{bmatrix} x_1 \\ nh \end{bmatrix}\right) - l_{mn}\left(\begin{bmatrix} x_1 \\ nh \end{bmatrix}\right) \right| \leq \frac{C}{8} h^2, \quad (44)$$

where  $l_{mn}$  is the linear interpolation function of  $f$  between points  $x = \begin{bmatrix} mh \\ nh \end{bmatrix}$  and  $x = \begin{bmatrix} mh+h \\ nh \end{bmatrix}$ , i.e.,

$$\begin{aligned} l_{mn}(x) &= \\ &\frac{(x_1 - mh)f\left(\begin{bmatrix} mh+h \\ nh \end{bmatrix}\right) + (mh+h - x_1)f\left(\begin{bmatrix} mh \\ nh \end{bmatrix}\right)}{h}. \end{aligned} \quad (45)$$

Thus,

$$\begin{aligned}
& \left| \int_{mh}^{mh+h} \left( f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 - \int_{mh}^{mh+h} \left( l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 \right| \\
&= \left| \int_{mh}^{mh+h} \left( f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) - l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 \right| \\
&\leq \int_{mh}^{mh+h} \left| f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) - l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right| dx_1 \\
&\leq \int_{mh}^{mh+h} \frac{C}{8} h^2 dx_1 = \frac{C}{8} h^3.
\end{aligned} \tag{46}$$

Besides, we have

$$\begin{aligned}
& \left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} f \left( \begin{bmatrix} mh \\ nh \end{bmatrix} \right) h \right| \\
&= \left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} \frac{h}{2} \left( f \left( \begin{bmatrix} mh \\ nh \end{bmatrix} \right) + f \left( \begin{bmatrix} mh+h \\ nh \end{bmatrix} \right) \right) \right| \\
&= \left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} \int_{mh}^{mh+h} \left( l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 \right| \\
&\leq \sum_{m \in \mathbb{Z}} \left| \int_{mh}^{mh+h} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \int_{mh}^{mh+h} \left( l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 \right|.
\end{aligned} \tag{47}$$

Since  $\forall \|x\| \geq (p+1/2)h$ ,  $\varphi(x) = 0$ , there are at most  $p+1$  values of  $m$ , s.t.,

$$\left| \int_{mh}^{mh+h} \left( f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 - \int_{mh}^{mh+h} \left( l_{mn} \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) \right) dx_1 \right| \neq 0, \tag{48}$$

while all of them satisfy (46). Then, by (46), (47) and (48), we can easily achieve (43).

2) We then prove

$$\begin{aligned}
& \left| \int_{\mathbb{R}^2} f(x) d\sigma(x) - \sum_{n \in \mathbb{Z}} \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 h \right| \\
&\leq \frac{(p+1)^2 C}{8} h^4.
\end{aligned} \tag{49}$$

Let  $F(x_2) = \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right) dx_1$ , and then

$$\int_{\mathbb{R}^2} f(x) d\sigma(x) = \int_{\mathbb{R}} F(x_2) dx_2. \tag{50}$$

By **Lemma 1**, we can obtain that  $|F''(x_2)| \leq (p+1)hC$ . Let  $L_n(x_2)$  be the linear interpolation function of  $F(x_2)$ , between points  $x_2 = nh$  and  $x_2 = nh+h$ , and then for any  $x_2 \in [nh, nh+h]$ , by **Lemma 2**, we have

$$|F(x_2) - L_n(x_2)| \leq \frac{(p+1)C}{8} h^3. \tag{51}$$

Thus, similar to (46), we can deduce that

$$\begin{aligned}
& \left| \int_{nh+h}^{nh} F(x_2) dx_2 - \int_{nh+h}^{nh} L_n(x_2) dx_2 \right| \\
&\leq \int_{nh+h}^{nh} |F(x_2) - L_n(x_2)| dx_2 \\
&\leq \frac{(p+1)C}{8} h^4.
\end{aligned} \tag{52}$$

Then, similar to (47), we can deduce that

$$\begin{aligned}
& \left| \int_{\mathbb{R}^2} f(x) d\sigma(x) - \sum_{n \in \mathbb{Z}} \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 h \right| \\
&= \left| \int_{\mathbb{R}} F(x_2) dx_2 - \sum_{n \in \mathbb{Z}} F(nh) h \right| \\
&= \left| \int_{\mathbb{R}} F(x_2) dx_2 - \sum_{n \in \mathbb{Z}} \frac{1}{2} (F(nh) + F(nh+h)) h \right| \\
&\leq \sum_{n \in \mathbb{Z}} \left| \int_{nh}^{nh+h} F(x_2) dx_2 - \frac{1}{2} (F(nh) + F(nh+h)) h \right|.
\end{aligned} \tag{53}$$

Since  $\forall \|x\| \geq (p+1/2)h$ ,  $\varphi(x) = 0$ , there are at most  $p+1$  values of  $m$ , s.t.,

$$\left| \int_{nh}^{nh+h} F(x_2) dx_2 - \frac{1}{2} (F(nh) + F(nh+h)) h \right| \neq 0, \tag{54}$$

while all of them satisfy (52). Then, by (52), (53) and (54), we can easily achieve (49).

3) By (43) and 49, we can obtain

$$\begin{aligned}
& \left| \int_{\mathbb{R}^2} \varphi(\tilde{A}^{-1}x) r(y-x) d\sigma(x) - \sum_{i \in \mathbb{E} \setminus \Lambda} \varphi(\tilde{A}^{-1}x_{ij}) r(y-x_{ij}) h^2 \right| \\
&= \left| \int_{\mathbb{R}} f(x) dx - \sum_{m,n \in \mathbb{Z}} f \left( \begin{bmatrix} mh \\ nh \end{bmatrix} \right) h^2 \right| \\
&\leq \left| \int_{\mathbb{R}} f(x) dx - \sum_{n \in \mathbb{Z}} \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 h \right| + \\
&\quad \sum_{n \in \mathbb{Z}} \left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) h \right| h,
\end{aligned} \tag{55}$$

where, by (49),

$$\begin{aligned}
& \left| \int_{\mathbb{R}} f(x) dx - \sum_{n \in \mathbb{Z}} \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 h \right| \\
&\leq \frac{(p+1)^2 C}{8} h^4.
\end{aligned} \tag{56}$$

Besides, Since  $\forall \|x\| \geq (p+1/2)h$ ,  $\varphi(x) = 0$ , we only have  $p+1$  values of  $m$ , s.t.,

$$\left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) h \right| \neq 0. \tag{57}$$

Thus, by (43), we have

$$\begin{aligned}
& \sum_{n \in \mathbb{Z}} \left| \int_{\mathbb{R}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) dx_1 - \sum_{m \in \mathbb{Z}} f \left( \begin{bmatrix} x_1 \\ nh \end{bmatrix} \right) h \right| h \\
&\leq \frac{(p+1)^2 C}{8} h^4.
\end{aligned} \tag{58}$$

Combining (55), (56), (58), we can deduce (42), which completes the proof.  $\square$

Finally, let us finally prove Theorem 1.

**Theorem 1.** Assume that an image  $I \in \mathbb{R}^{n \times n}$  is discretized from the smooth function  $r : \mathbb{R}^2 \rightarrow \mathbb{R}$  by (22), a feature map  $F \in \mathbb{R}^{n \times n \times t}$  is discretized from the smooth function  $e : \mathbb{R}^2 \times S \rightarrow \mathbb{R}$  by (23),  $|S| = t$ , and filters  $\tilde{\Psi}$ ,  $\tilde{\Phi}$  and  $\tilde{\Upsilon}$  are generated from

$\varphi_{in}$ ,  $\varphi_{out}$  and  $\varphi_A$ ,  $\forall A \in S$ , by (24), respectively. If for any  $A \in S$ ,  $x \in \mathbb{R}^2$ , the following conditions are satisfied:

$$\begin{aligned} & |r(x)|, |e(x, A)| \leq F_1, \\ & \|\nabla r(x)\|, \|\nabla e(x, A)\| \leq G_1, \\ & \|\nabla^2 r(x)\|, \|\nabla^2 e(x, A)\| \leq H_1, \\ & |\varphi_{in}(x)|, |\varphi_A(x)|, |\varphi_{out}(x)| \leq F_2, \\ & \|\nabla \varphi_{in}(x)\|, \|\nabla \varphi_A(x)\|, \|\nabla \varphi_{out}(x)\| \leq G_2, \\ & \|\nabla^2 \varphi_{in}(x)\|, \|\nabla^2 \varphi_A(x)\|, \|\nabla^2 \varphi_{out}(x)\| \leq H_2, \\ & \forall \|x\| \geq (p+1)h/2, \varphi_{in}(x), \varphi_A(x), \varphi_{out}(x) = 0, \end{aligned} \quad (59)$$

where  $p$  is the filter size,  $h$  is the mesh size, and  $\nabla$  and  $\nabla^2$  denote the operators of gradient and Hessian matrix, respectively, then for any  $\tilde{A} \in S$ , the following results are satisfied:

$$\begin{aligned} & \left\| \tilde{\Psi} \star \tilde{\pi}_{\tilde{A}}^R(I) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Psi} \star I) \right\|_{\infty} \leq \frac{C}{2}(p+1)^2 h^2, \\ & \left\| \tilde{\Phi} \star \tilde{\pi}_{\tilde{A}}^E(F) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Phi} \star F) \right\|_{\infty} \leq \frac{C}{2}(p+1)^2 h^2 t, \\ & \left\| \tilde{\Upsilon} \star \tilde{\pi}_{\tilde{A}}^E(F) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Upsilon} \star F) \right\|_{\infty} \leq \frac{C}{2}(p+1)^2 h^2 t, \end{aligned} \quad (60)$$

where  $C = F_1 H_2 + F_2 H_1 + 2G_1 G_2$ ,  $\tilde{\pi}_{\tilde{A}}^R$ ,  $\tilde{\pi}_{\tilde{A}}^E$ ,  $\tilde{\Psi}$ ,  $\tilde{\Phi}$  and  $\tilde{\Upsilon}$  are defined by (24) and (29), respectively, the operators  $\star$  involved in Eq. (60) are defined in (26), (27) and (28), respectively, and  $\|\cdot\|_{\infty}$  represents the infinity norm.

*Proof.* For any  $y \in \mathbb{R}$ ,  $A, B \in S$ , let

$$\hat{\Psi}[r](y, A) = \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{in}(A^{-1}x_{\tilde{i}\tilde{j}}) r(y - x_{\tilde{i}, \tilde{j}}), \quad (61)$$

where  $\Lambda = \{(\tilde{i}, \tilde{j}) | \tilde{i}, \tilde{j} = 1, 2, \dots, p\}$ . Then, for any  $A \in S$ , we can obtain

$$\hat{\Psi}[r](y_{ij}, A) = (\hat{\Psi} \star I)_{ij}^A. \quad (62)$$

1) By **Remark 2**, we know that  $\hat{\Psi}[\pi_{\tilde{A}}^R[r]] = \pi_{\tilde{A}}^E[\Psi[r]]$ . Thus for any  $A \in S$ , we have

$$\begin{aligned} & \left| (\tilde{\Psi} \star \tilde{\pi}_{\tilde{A}}^R(I) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Psi} \star I))_{ij}^A \right| \\ &= \left| \hat{\Psi}[\pi_{\tilde{A}}^R[r]](y_{ij}, A) - \pi_{\tilde{A}}^E[\hat{\Psi}[r]](y_{ij}, A) \right| \\ &\leq \left| \hat{\Psi}[\pi_{\tilde{A}}^R[r]](y_{ij}, A) - \frac{1}{h^2} \Psi[\pi_{\tilde{A}}^R[r]](y_{ij}, A) \right| \\ &\quad + \left| \pi_{\tilde{A}}^E[\hat{\Psi}[r]](y_{ij}, A) - \frac{1}{h^2} \pi_{\tilde{A}}^E[\Psi[r]](y_{ij}, A) \right|. \end{aligned} \quad (63)$$

Let  $\hat{r} = \pi_{\tilde{A}}^R[r]$ , and then it is easy to deduce that  $\hat{r}$  satisfies the conditions in **Lemma 3**. Then, by **Lemma 3**,

$$\begin{aligned} & \left| \hat{\Psi}[\pi_{\tilde{A}}^R[r]](y_{ij}, A) - \frac{1}{h^2} \Psi[\pi_{\tilde{A}}^R[r]](y_{ij}, A) \right| \\ &= \frac{1}{h^2} \left| \hat{\Psi}[\pi_{\tilde{A}}^R[r]](y_{ij}, A) h^2 - \Psi[\pi_{\tilde{A}}^R[r]](y_{ij}, A) \right| \\ &= \frac{1}{h^2} \left| \sum_{(i,j) \in \Lambda} \varphi_{in}(A^{-1}x_{ij}) \hat{r}(y_{ij} - x_{i,j}) h^2 - \right. \\ &\quad \left. \int_{\mathbb{R}^2} \varphi_{in}(A^{-1}x) \hat{r}(y_{ij} - x) d\sigma(x) \right| \\ &\leq \frac{(p+1)^2 C}{4} h^2. \end{aligned} \quad (64)$$

Besides, let  $\hat{A} = \tilde{A}^{-1}A$  and  $\hat{y}_{ij} = \tilde{A}^{-1}y_{ij}$ , and by **Lemma 3**, we can also achieve,

$$\begin{aligned} & \left| \pi_{\tilde{A}}^E[\hat{\Psi}[r]](y_{ij}, A) - \frac{1}{h^2} \pi_{\tilde{A}}^E[\Psi[r]](y_{ij}, A) \right| \\ &= \frac{1}{h^2} \left| \pi_{\tilde{A}}^E[\hat{\Psi}[r]](y_{ij}, A) h^2 - \pi_{\tilde{A}}^E[\Psi[r]](y_{ij}, A) \right| \\ &= \frac{1}{h^2} \left| \sum_{(i,j) \in \Lambda} \varphi_{in}(\hat{A}^{-1}x_{ij}) r(\hat{y}_{ij} - x_{i,j}) h^2 - \right. \\ &\quad \left. \int_{\mathbb{R}^2} \varphi_{in}(\hat{A}^{-1}x) r(\hat{y}_{ij} - x) d\sigma(x) \right| \\ &\leq \frac{(p+1)^2 C}{4} h^2. \end{aligned} \quad (65)$$

Thus, combining (63), (64) and (65), we can achieve

$$\left| \hat{\Psi}[\pi_{\tilde{A}}^R[r]](y_{ij}, A) - \pi_{\tilde{A}}^E[\hat{\Psi}[r]](y_{ij}, A) \right| \leq \frac{C}{2}(p+1)^2 h^2. \quad (66)$$

In other word,

$$\left| \left( \tilde{\Psi} \star \tilde{\pi}_{\tilde{A}}^R(I) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Psi} \star I) \right)_{ij}^A \right| \leq \frac{C}{2}(p+1)^2 h^2. \quad (67)$$

This proves the first inequality in (60).

2) For any  $A, B \in S$ , let  $\hat{B} = \tilde{A}^{-1}B$ ,  $r_A = e(x, A)$ , and  $\hat{\Psi}_A$  be a operator defined in the formulation of (61), while correlated to  $\varphi_A$ . Then, for any  $i, j = 1, 2, \dots, n$ ,  $B \in S$ ,

$$\begin{aligned} & \left| \left( \tilde{\Phi} \star \tilde{\pi}_{\tilde{A}}^E(F) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Phi} \star F) \right)_{ij}^B \right| \\ &= \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, A \in S} \varphi_A(B^{-1}x_{\tilde{i}\tilde{j}}) e(\tilde{A}^{-1}(y_{ij} - x_{\tilde{i}, \tilde{j}}), \tilde{A}^{-1}BA) - \right. \\ &\quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, A \in S} \varphi_A(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) e(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}, \tilde{A}^{-1}BA) \right| \\ &\leq \sum_{A \in S} \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_A(B^{-1}x_{\tilde{i}\tilde{j}}) r_{\hat{B}A}(\tilde{A}^{-1}y_{ij} - \tilde{A}^{-1}x_{\tilde{i}, \tilde{j}}) - \right. \\ &\quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_A(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) r_{\hat{B}A}(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}) \right| \\ &= \sum_{A \in S} \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_A(B^{-1}x_{\tilde{i}\tilde{j}}) \pi_{\tilde{A}}^R[r_{\hat{B}A}](y_{ij} - x_{\tilde{i}, \tilde{j}}) - \right. \\ &\quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_A(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) r_{\hat{B}A}(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}) \right| \\ &= \sum_{A \in S} \left| \hat{\Psi}_A[\pi_{\tilde{A}}^R[r_{\hat{B}A}]](y_{ij}, B) - \pi_{\tilde{A}}^E[\hat{\Psi}_A[r_{\hat{B}A}]](y_{ij}, B) \right|. \end{aligned}$$

Then by (66), we can achieve that  $\forall i, j = 1, 2, \dots, n$ ,  $B \in S$ ,

$$\left| \left( \tilde{\Phi} \star \tilde{\pi}_{\tilde{A}}^E(F) - \tilde{\pi}_{\tilde{A}}^E(\tilde{\Phi} \star F) \right)_{ij}^B \right| \leq \frac{C}{2}(p+1)^2 h^2 t. \quad (68)$$

This proves the second inequality in (60).

3) For any  $A, B \in S$ , let  $\hat{B} = \tilde{A}^{-1}B$ ,  $r_A = e(x, A)$ , and  $\hat{\Psi}_{out}$  be a operator defined in the formulation of (61), while correlated to  $\varphi_{out}$ .

Then, we have that  $\forall i, j = 1, 2, \dots, n$ ,

$$\begin{aligned} & \left| \left( \tilde{\Upsilon} * \tilde{\pi}_{\tilde{A}}^{\tilde{E}}(F) - \pi_{\tilde{A}}^{\tilde{E}}(\tilde{\Upsilon} * F) \right)_{ij} \right| \\ &= \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, B \in S} \varphi_{out}(B^{-1}x_{\tilde{i}\tilde{j}}) e\left(\tilde{A}^{-1}(y_{ij} - x_{\tilde{i}, \tilde{j}}), \tilde{A}^{-1}B\right) - \right. \\ & \quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda, B \in S} \varphi_{out}(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) e\left(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}, \tilde{A}^{-1}B\right) \right| \\ &\leq \sum_{B \in S} \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{out}(B^{-1}x_{\tilde{i}\tilde{j}}) r_{\hat{B}}\left(\tilde{A}^{-1}y_{ij} - \tilde{A}^{-1}x_{\tilde{i}, \tilde{j}}\right) - \right. \\ & \quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{out}(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) r_{\hat{B}}\left(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}, \tilde{A}^{-1}B\right) \right| \\ &= \sum_{B \in S} \left| \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{out}(B^{-1}x_{\tilde{i}\tilde{j}}) \pi_{\tilde{A}}^R[r_{\hat{B}}](y_{ij} - x_{\tilde{i}, \tilde{j}}) - \right. \\ & \quad \left. \sum_{(\tilde{i}, \tilde{j}) \in \Lambda} \varphi_{out}(B^{-1}\tilde{A}x_{\tilde{i}\tilde{j}}) r_{\hat{B}}\left(\tilde{A}^{-1}y_{ij} - x_{\tilde{i}, \tilde{j}}\right) \right| \\ &= \sum_{B \in S} \left| \hat{\Psi}_{out}\left[\pi_{\tilde{A}}^R[r_{\hat{B}}]\right](y_{ij}, B) - \pi_{\tilde{A}}^E\left[\hat{\Psi}_{out}[r_{\hat{B}}]\right](y_{ij}, B) \right|. \end{aligned}$$

Then by (66), we can achieve that  $\forall i, j = 1, 2, \dots, n$ ,

$$\left| \left( \tilde{\Upsilon} * \tilde{\pi}_{\tilde{A}}^{\tilde{E}}(F) - \pi_{\tilde{A}}^{\tilde{E}}(\tilde{\Upsilon} * F) \right)_{ij} \right| \leq \frac{C}{2}(p+1)^2 h^2 t. \quad (69)$$

This proves the third inequality in (60).  $\square$

## 2 MORE EXPLORATIONS ON FOURIER BASES

### 2.1 Aliasing Effect of Fourier Bases

Here we shortly introduce the aliasing effect of Fourier Bases.

Let us have a look at the 1D example depicted in Fig. 1 for easily understanding the aliasing effect. From the figure, we can see that the discretizations of a high frequency and a low frequency cosine function can be exactly the same. However, the discretization of the transformation results of the high frequency one tend to be highly unpredictable.

Such aliasing effect also easily occurs in the commonly used 2D Fourier series expansion. As depicted by the basis in Fig. 1 (c), 2D Fourier bases with high frequency look similar to that with low frequency shown in Fig. 1 (b) when there are no rotations. However, the high frequency basis will be badly destructed after rotation.

### 2.2 Other Explorations

Firstly, we can deduce the following conclusion.

**Remark 3.** When  $p$  is an odd number, the basis set defined in Eq. (2) can be equivalently reduced to  $p^2$  elements. When  $p$  is an even number, the basis set defined in Eq. (2) can be equivalently reduced to  $(p+1)^2 - 2$  elements.

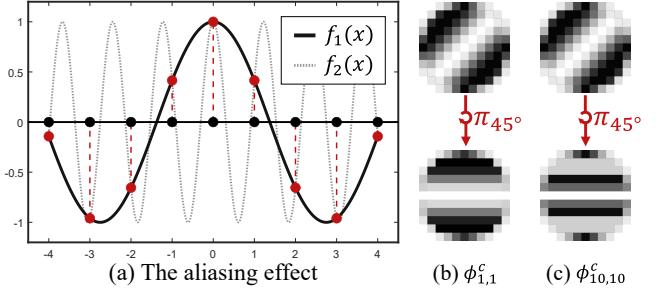


Fig. 1. (a) An 1D example showing the insight under this aliasing effect:  $\forall x \in \mathbb{N}$ ,  $f_1(x) = f_2(x)$ , where  $f_1(x) = \cos(2\pi/p \cdot 2x)$ ,  $f_2(x) = \cos(2\pi/p \cdot 9x)$ . Although the discretization of the two functions are the same, interpolation of the high frequency one tends to be highly unpredictable. (b)-(c) illustration of rotating  $\phi_{1,1}^c$  and  $\phi_{10,10}^c$  by  $45^\circ$ . Note that  $\phi_{1,1}^c$  is the same as  $\phi_{10,10}^c$ , but their rotation results are different to each other, which is due to the heavy aliasing effect in  $\phi_{10,10}^c$ .

*Proof.* 1) When  $p$  is an odd number, we can deduce that  $\forall k, l = 0, 1, \dots, p-1$ ,

$$\begin{aligned} \varphi_{kl}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(k - \left\lfloor \frac{p}{2} \right\rfloor\right)x_1 + \left(l - \left\lfloor \frac{p}{2} \right\rfloor\right)x_2\right)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(\left\lfloor \frac{p}{2} \right\rfloor - k\right)x_1 + \left(\left\lfloor \frac{p}{2} \right\rfloor - l\right)x_2\right)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(p-1-k-\left\lfloor \frac{p}{2} \right\rfloor\right)x_1 + \left(p-1-l-\left\lfloor \frac{p}{2} \right\rfloor\right)x_2\right)\right) \\ &= \varphi_{(p-1-k)(p-1-l)}^c(x). \end{aligned} \quad (70)$$

Similarly, we also can deduce

$$\varphi_{kl}^s(x) = -\varphi_{(p-1-k),(p-1-l)}^s(x). \quad (71)$$

Besides, it is easy to deduce that  $0 \leq (p-1-k) \leq p-1$ ,  $0 \leq (p-1-l) \leq p-1$ . We also have  $(k, l) = (p-1-k, p-1-k)$  if and only if  $(k, l) = (\lfloor p/2 \rfloor, \lfloor p/2 \rfloor)$ . This means that we can reduce half of bases in the case  $(k, l) \neq (\lfloor p/2 \rfloor, \lfloor p/2 \rfloor)$ , i.e.,  $p^2 - 1$  bases are enough.

Moreover in the case  $(k, l) = (\lfloor p/2 \rfloor, \lfloor p/2 \rfloor)$ , there are only two bases,  $\varphi_{kl}^c(x) = \cos(0) = 1$  and  $\varphi_{kl}^s(x) = \sin(0) = 0$ , where we only need  $\varphi_{kl}^c(x)$  when performing filter parametrization.

In summary, the proposed basis set actually can be reduced to  $p^2$  elements.

2) When  $p$  is an even number, we can deduce that

$$\begin{aligned} \varphi_{kl}^c(x) &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(k - \left\lfloor \frac{p}{2} \right\rfloor\right)x_1 + \left(l - \left\lfloor \frac{p}{2} \right\rfloor\right)x_2\right)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(\left\lfloor \frac{p}{2} \right\rfloor - k\right)x_1 + \left(\left\lfloor \frac{p}{2} \right\rfloor - l\right)x_2\right)\right) \\ &= \Omega(x) \cos\left(\frac{2\pi}{ph}\left(\left(p-k-\left\lfloor \frac{p}{2} \right\rfloor\right)x_1 + \left(p-l-\left\lfloor \frac{p}{2} \right\rfloor\right)x_2\right)\right) \\ &= \varphi_{(p-k)(p-l)}^c(x). \end{aligned} \quad (72)$$

Similarly, we also can deduce

$$\varphi_{kl}^s(x) = -\varphi_{(p-k),(p-l)}^s(x). \quad (73)$$

It is easy to obtain that for the case when  $k, l = 1, 2, \dots, p-1$  and  $(k, l) \neq (\lfloor p/2 \rfloor, \lfloor p/2 \rfloor)$ , we have  $((p-k), (p-l)) \neq (k, l)$

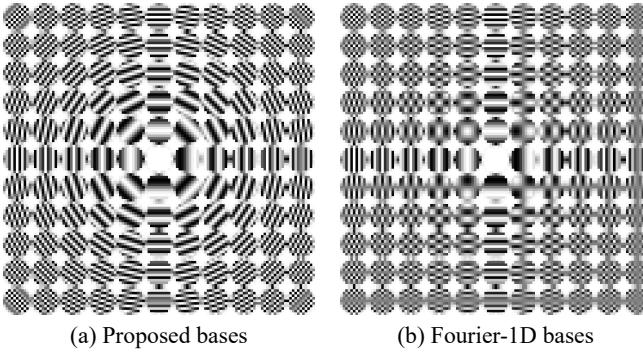


Fig. 2. Illustrations of (a) the proposed bases defined in Eq. (74) and (b) 1D Fourier bases defined in Eq. (75), where  $p = 11$ . We have removed the duplicates in the proposed basis set and only left  $p^2$  elements for better visualization.

and  $1 \leq (p-k) \leq p-1, 1 \leq (p-l) \leq p-1$ . This means that we can reduce half of the bases in this case, where the total number of bases is  $2(p-1)^2 - 2$ . In other word,  $(p-1)^2 - 1$  bases are enough in this case.

Besides, in the case  $(k, l) = ([p/2], [p/2])$ , there are only two bases,  $\varphi_{kl}^c(x) = \cos(0) = 1$  and  $\varphi_{kl}^s(x) = \sin(0) = 0$ , where we only need  $\varphi_{kl}^c(x)$  when performing filter parametrization.

In the case otherwise the above two cases (i.e.,  $k = 0$  cases and  $l = 0$  cases), the total basis number is  $4p - 2$ . Therefore, the number of bases we need is  $(p-1)^2 - 1 + 1 + 4p - 2 = (p+1)^2 - 2$ .  $\square$

From Remark 3, we know that about half number of the elements in the proposed basis set is enough for our filter parametrization. More specifically, when  $p$  is a odd number, the proposed basis set defined in Eq. (2) is equivariant to,

$$\begin{aligned} \varphi_{kl}^c(x) &= \Omega(x) \cos \left( \frac{2\pi}{ph} \left[ k - \left\lfloor \frac{p}{2} \right\rfloor, l - \left\lfloor \frac{p}{2} \right\rfloor \right] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right), \\ \forall k, l \in \{0 \leq k \leq [p/2], 0 \leq l \leq [p/2]\} \text{ and} \\ \forall k, l \in \{0 \leq k < [p/2], [p/2] < l \leq p-1\}; \\ \varphi_{kl}^s(x) &= \Omega(x) \sin \left( \frac{2\pi}{ph} \left[ k - \left\lfloor \frac{p}{2} \right\rfloor, l - \left\lfloor \frac{p}{2} \right\rfloor \right] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right), \\ \forall k, l \in \{0 \leq k \leq [p/2], [p/2] < l \leq p-1\} \text{ and} \\ \forall k, l \in \{0 \leq k < [p/2], 0 \leq l \leq [p/2]\}. \end{aligned} \quad (74)$$

One can refer to Fig. 2 for an intuitive understanding of this basis set.

Since  $\cos(\alpha + \beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$  and  $\sin(\alpha + \beta) = \cos(\alpha)\sin(\beta) + \sin(\alpha)\cos(\beta)$ , it is naturally to conduct another basis set expressed as follows:

$$\begin{aligned} \varphi_{kl}^{cc}(x) &= \cos \left( \frac{2\pi}{ph} kx \right) \cos \left( \frac{2\pi}{ph} lx \right), \quad k, l = 0, 1, \dots, [p/2]; \\ \varphi_{kl}^{cs}(x) &= \cos \left( \frac{2\pi}{ph} kx \right) \sin \left( \frac{2\pi}{ph} lx \right), \quad k = 0, 1, \dots, [p/2], \\ &\quad l = 1, 2, \dots, [p/2]; \\ \varphi_{kl}^{sc}(x) &= \sin \left( \frac{2\pi}{ph} kx \right) \cos \left( \frac{2\pi}{ph} lx \right), \quad k = 1, 2, \dots, [p/2], \\ &\quad l = 0, 1, \dots, [p/2]; \\ \varphi_{kl}^{ss}(x) &= \sin \left( \frac{2\pi}{ph} kx \right) \sin \left( \frac{2\pi}{ph} lx \right), \quad k, l = 1, 2, \dots, [p/2]. \end{aligned} \quad (75)$$

We call this basis set as Fourier-1D basis in this paper, and It is easy to deduce that when  $p$  is a odd number, this basis set contains  $p^2$  elements, and can represent all elements in the proposed basis set, vice versa. When  $p$  is an even number, this basis set contains  $(p+1)^2$  elements and can represent all elements in the proposed bases, where the proposed basis set can represent almost all the elements in this basis set. Therefore, the bases in Eq. (75) are actually with similar expression ability as the proposed bases, while its highest frequency is also  $[p/2]$ . It is thus also a good choice for constructing filter parametrization.

As shown in the following experimental results, Fourier-1D bases can achieve comparable performance with the proposed ones in low-level tasks like image super resolution, while its performance would be a little lower than the proposed bases in the MNIST-rot classification tasks. Besides, from Fig. 2, we can observe that the proposed basis set is closer to the texture natural image than Fourier-1D one, in the sense of human intuition. Therefore we still prefer to adopt the proposed basis set for filter parametrization in the main paper.

### 2.3 More Bandlimit Analysis

We have carefully tuned the band-limiting (by removing bases with high frequency) for the Fourier bases, and it turns out that the bandlimited bases we used (case 3 in Fig. 3) can help achieve the best performance. In Table 2.3, we show filer parametrization results of the full-band Fourier bases, 3 different band-limited versions of Fourier bases, and the proposed bases, where the bases of the 3 band-limited versions are shown in Fig. 3 (b)(c) and (d), respectively. It can be observed that the full band Fourier bases and the proposed ones can achieve the same representation accuracy for a filter, while band-limitation hampers the representation accuracy. However, when rotating the filter parametrization results for  $45^\circ$ , the rotation results of case 3 can achieve better result than case 1, case2 and the full-band cases. Besides, when expressing  $0^\circ$  and  $45^\circ$  filters simultaneously, the band-limited Fourier bases case achieves the same representation accuracy as full band cases, and outperforms case 1 and case 2 with less band-limitation. This can be rationally explained by the fact that the discretizations of high frequency bases incline to be very different in  $0^\circ$  and  $45^\circ$ , which tends to make the high frequency bases hardly be essentially used in representing the object in  $0^\circ$  and  $45^\circ$  simultaneously. Comparatively, band-limiting of case 3 will have less effect on the representation accuracy when expressing under different orientations.

It is indeed a limitation of the original Fourier bases that its band limitation would cause the losing of orientations. However, the band-limiting of the proposed bases would alleviate this problem as shown in Fig. 4.

## 3 MORE IMPLEMENTATION DETAILS

In the definition of the proposed bases (2), we exploit a circular radial mask function  $\Omega(x)$  to limit the range of bases, and make them more suitable for rotation. The mask

TABLE 1

The RMSE (mean $\pm$ standard deviation over 1000 random generated samples) of filter parametrization on continuous functions, obtained by all competing methods under different filter sizes.

Method	parameterization before rotation			simultaneously
	original	45° rotation	0° & 45°	
Fourier (Full bands)	<b>9.7e-13±3.8e-14</b>	9.0e-01±2.4e-01	3.6e-01±2.1e-01	
Fourier (case 1)	1.5e-12±4.3e-13	8.5e-01±3.7e-01	3.9e-01±2.5e-01	
Fourier (case 2)	2.4e-01±3.0e-01	6.1e+00±8.8e+00	4.0e-01±4.0e-01	
Fourier (case 3)	4.2e-01±4.2e-01	<u>4.4e-01±4.2e-01</u>	3.6e-01±3.5e-01	
Proposed	<b>9.7e-13±3.8e-14</b>	<b>4.1e-02±9.2e-03</b>	<b>2.0e-02±2.4e-03</b>	

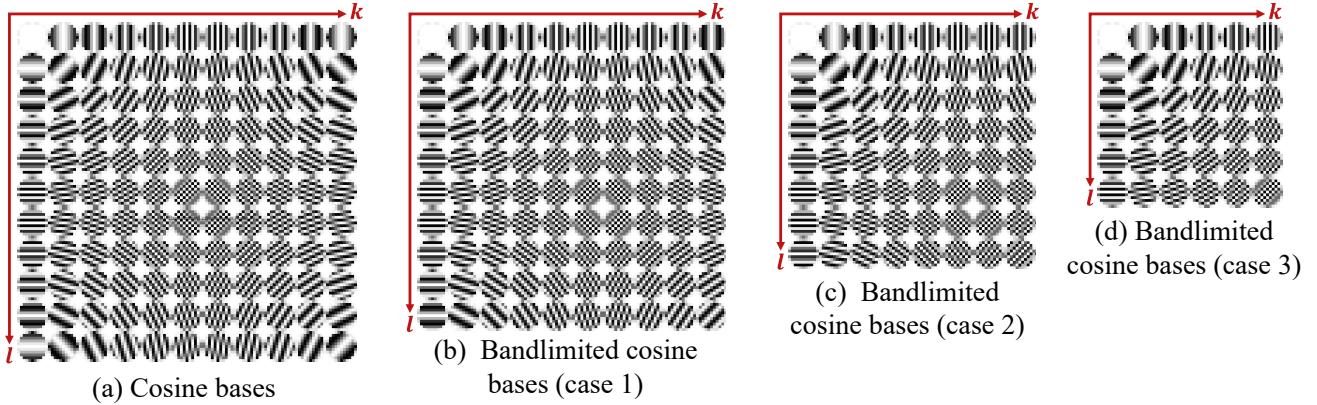


Fig. 3. (a) illustrations of the Fourier bases, where we take cosine bases as examples. (b) (c) and (d) are illustrations of 3 band-limited version of Fourier bases.

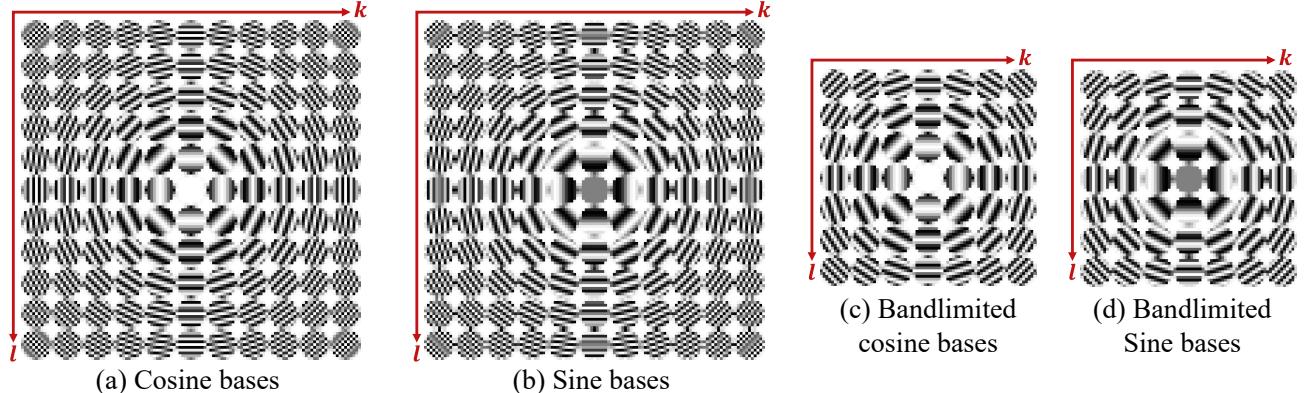


Fig. 4. (a) and (b) are illustrations of the proposed bases. (c) and (d) are illustrations of the band-limitation of the proposed bases.

satisfies that  $\Omega(x) = 0$  if  $\|x\| \geq (p+1/2)h$ . Specifically, we set it as:

$$\Omega(x) = \begin{cases} 0 & \text{if } \|x\| \geq (r+h) \\ e^{-5 \cdot \max\left\{\frac{\|x\|^2}{r^2} - 1, 0\right\}} & \text{if } \|x\| < (r+h) \end{cases}, \quad (76)$$

where  $r = (p-1/2)h$ . It is easy to deduce that when  $\|x\| < r$ ,  $\Omega(x) = 1$ , and  $\Omega(x)$  is in a circular shape with soft edge. Fig. 5 is an illustration of the exploited  $\Omega(x)$ , where one can intuitively observe that when the larger the filter size  $p$  is, the more circular the mask is.

## 4 MORE EXPERIMENTAL RESULTS

In this section, we show more experimental results about the utilized bases, the proposed filter parametrization, and the proposed equivariant convolutions, respectively.

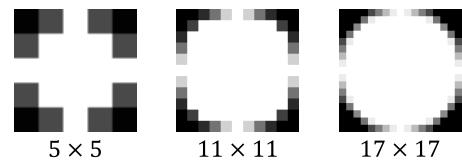


Fig. 5. Illustration of the circular radial mask function  $\Omega(x)$ . From left to right: illustrations of  $\Omega(x)$  discretized with filter size  $p = 5$ ,  $p = 11$  and  $p = 17$ , respectively.

### 4.1 Rotation of Bases

Alleviating the aliasing effect when rotating to an arbitrary angle is the most important advantage of the proposed filter parametrization method, as compared to the original 2D Fourier series expansion. In Fig. 3 of the main text, we have shown the superiority of the proposed bases as compared to traditional 2D Fourier bases. Here we provide more related

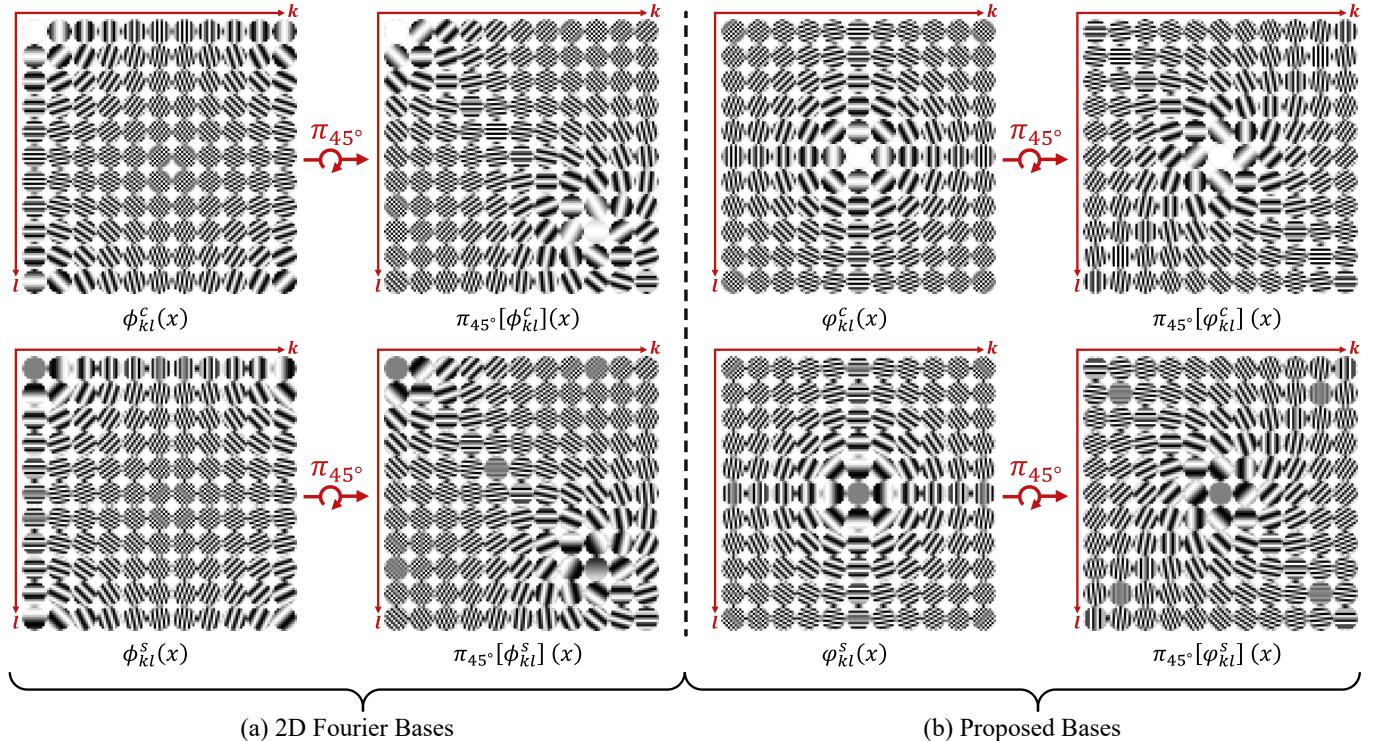


Fig. 6. (a) Illustrations of the 2D Fourier bases,  $\phi_{kl}^c$  and  $\phi_{kl}^s$ , and their  $45^\circ$  rotations (rotated by  $\pi_{45^\circ}(\cdot)$ ), where  $k, l = 0, 1, \dots, p - 1$  and  $p = 11$ . The aliasing effects can be observed in most of the bases, which also result in an unexpected vortex-like pattern in the maps of all these bases. (b) Illustrations of the proposed bases,  $\varphi_{kl}^c$  and  $\varphi_{kl}^s$ , and their  $45^\circ$  rotations, where  $k, l = 0, 1, \dots, p - 1$  and  $p = 11$ . The aliasing effect is obviously alleviated to a large extent.

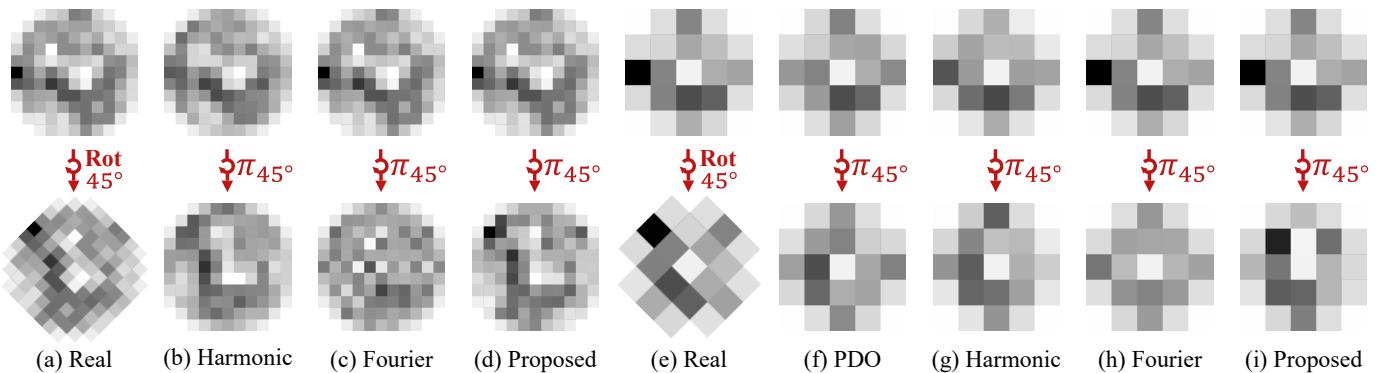


Fig. 7. (a) A discrete random filter generated by (30) of the main text, and its  $\pi/4$  rotation in visualization, with filter size  $p = 11$ . (b)-(d) The representations and correlated  $\pi/4$  rotations of a given 2D filter, where the harmonics bases [1], 2D Fourier bases [2] and the proposed bases in this study are adopted as basis functions, respectively. (e) A discrete random filter generated by (30) of the main text and its  $\pi/4$  rotation in visual, with filter size  $p = 5$ . (f)-(i) The representations and correlated  $\pi/4$  rotations of a given 2D filter, by adopting PDO bases [3], harmonics bases [1], 2D Fourier bases [2] and the proposed bases in this study as basis functions, respectively.

demonstrations on this point.

In Fig. 6, we show the rotation of all the proposed bases ( $\varphi_{kl}^c$  and  $\varphi_{kl}^s$ ,  $k, l = 1, 2, \dots, p - 1$ , defined in (2)) as compared to the traditional 2D Fourier bases ( $\phi_{kl}^c$  and  $\phi_{kl}^s$ ,  $k, l = 1, 2, \dots, p - 1$ , defined in (1)), where the patch size  $p = 11$ . From Fig. 6(a), we can see that, when being clockwise rotated with  $45^\circ$ , the high frequency ones of 2D Fourier bases suffer evident aliasing effect issue, i.e., many of bases are completely distorted into other shapes, which even results in an unexpected vortex-like pattern in the map consisted by all bases. Comparatively, the proposed bases tends to largely alleviate such aliasing phenomenon in most

of the bases, i.e., most of the bases are able to keep their shapes after rotation, and the vortex-like pattern rarely arises.

## 4.2 More Filter Parametrization Verifications

**Visual results on random initialization.** In the main text we have shown the illustration of filter Parametrization result on a continuous function. Here, we further provide the visual results about Parametrization of random initialization. Fig. 7 shows an example of the representation results of Eq. (30) in the main text for the cases  $p = 11$  and  $p = 5$ , respectively, where PDO based method is only tested on the case  $p = 5$ . In this experiment, although there is no ground

TABLE 2

The RMSE (mean $\pm$ standard deviation over 1000 random generated samples) of filter parametrization on continuous functions, obtained by all competing methods under different filter sizes.

Method	11 × 11			5 × 5		
	parameterization before rotation		simultaneously	parameterization before rotation		simultaneously
	original	45° rotation	0° & 45°	original	45° rotation	0° & 45°
PDO	-	-	-	4.3e-01 $\pm$ 6.5e-02	1.1e-01 $\pm$ 1.1e-01	2.6e-01 $\pm$ 3.3e-02
Harmonic	3.5e-01 $\pm$ 4.3e-02	3.7e-01 $\pm$ 4.5e-02	2.7e-01 $\pm$ 3.4e-02	4.6e-01 $\pm$ 6.5e-02	1.3e-01 $\pm$ 1.3e-01	2.7e-01 $\pm$ 3.4e-02
Harmonic+	1.3e-01 $\pm$ 3.3e-02	1.6e-01 $\pm$ 2.4e-02	1.2e-01 $\pm$ 2.5e-02	6.2e-02 $\pm$ 1.5e-02	6.5e-02 $\pm$ 6.5e-02	1.2e-01 $\pm$ 2.5e-02
Fourier	<b>9.7e-13<math>\pm</math>4.1e-14</b>	9.1e-01 $\pm$ 2.4e-01	3.7e-01 $\pm$ 2.2e-01	<b>1.7e-10<math>\pm</math>3.0e-11</b>	2.1e-01 $\pm$ 2.1e-01	3.7e-01 $\pm$ 2.2e-01
Fourier+	4.2e-01 $\pm$ 4.1e-01	4.4e-01 $\pm$ 4.2e-01	3.6e-01 $\pm$ 3.5e-01	1.5e-01 $\pm$ 1.2e-01	4.3e-01 $\pm$ 4.3e-01	3.6e-01 $\pm$ 3.5e-01
Fourier-1D	<b>9.7e-13<math>\pm</math>4.2e-14</b>	<b>4.1e-02<math>\pm</math>9.7e-03</b>	<b>2.0e-02<math>\pm</math>2.6e-03</b>	<b>1.7e-10<math>\pm</math>3.0e-11</b>	<b>2.4e-02<math>\pm</math>2.4e-02</b>	<b>6.6e-02<math>\pm</math>2.6e-03</b>
Proposed	<b>9.7e-13<math>\pm</math>4.2e-14</b>	<b>4.1e-02<math>\pm</math>9.7e-03</b>	<b>2.0e-02<math>\pm</math>2.6e-03</b>	<b>1.7e-10<math>\pm</math>3.0e-11</b>	<b>2.4e-02<math>\pm</math>2.4e-02</b>	<b>6.6e-02<math>\pm</math>2.6e-03</b>

TABLE 3

The RMSE (mean $\pm$ standard deviation over 1000 random generated samples) of filter parametrization on random initialization filters, obtained by all competing methods under different filter sizes.

Method	11 × 11		5 × 5	
	original	45° rotation	original	45° rotation
PDO	-	-	4.9e-01 $\pm$ 1.3e-01	-
Harmonic	4.2e-01 $\pm$ 5.9e-02	-	5.7e-01 $\pm$ 1.2e-01	-
Harmonic+	2.0e-01 $\pm$ 3.8e-02	-	1.8e-01 $\pm$ 5.9e-02	-
Fourier	<b>9.5e-13<math>\pm</math>3.8e-14</b>	-	<b>6.2e-11<math>\pm</math>2.3e-11</b>	-
Fourier+	5.0e-01 $\pm$ 7.1e-02	-	2.0e-01 $\pm$ 7.7e-02	-
Fourier-1D	<b>9.5e-13<math>\pm</math>5.8e-14</b>	-	8.6e-11 $\pm$ 2.9e-11	-
Proposed	<b>9.5e-13<math>\pm</math>3.8e-14</b>	-	<b>6.2e-11<math>\pm</math>2.3e-11</b>	-

truth for the rotation of the object filter, the performance of rotation can still be visually assessed through observation. For the case  $p = 11$ , it is easy to see that the filter representation result of harmonics bases is worse than other two comparison methods, where some sharp dark and bright pixels are evidently lost, also leading to a blurry effect of the rotation result. Meanwhile, the rotation result achieved by classical Fourier bases is obviously harmed by the aliasing effect, although its representation of the original input is exact. As comparison, the proposed method can achieve relatively better representation and rotation for a random filter. For the case of  $p = 5$ , one can also easily observe that the proposed method outperforms other competing methods. In particular, the filter representations of PDO and harmonics based method are more degenerated among all comparison methods, and the proposed method more faithfully preserves the local configurations (e.g., the black in the upper left and the deep gray in the upper right) after filter rotation.

**Comparison with more filter parametrization methods.** In the main text, we have provided experimental results on the proposed filter parametrization method, with comparison to traditional 2D-Fourier-bases-based filter parametrization (Fourier), the harmonics-based filter parametrization (harmonic, [1], [4]) and partial-differential-operator-based filter parametrization (PDO, [3]). In this section, We provide more experimental result for verifying the proposed filter parametrization method.

Since the heavy aliasing effect in the Fourier basis set is conducted due to the high frequency bases, it is natural to wonder whether the proposed basis set is better than a properly bandlimited Fourier bases in the sense of filter parametrization. Therefore, we build the Fourier+ method,

which is the filer parametrization method based on Fourier bases with carefully tuned bandlimiting.

Besides, the harmonics used in [1] and [4] are heavily bandlimited for more stability during rotations, which tend to hamper the expression ability. In this section, we conduct a Harmonic+ filter parametrization method, which is based on the harmonic bases with less bandlimiting, where we carefully tune the parameters of harmonic bases ( $\tau$  and  $K_j$  for  $j = 1, \dots, J$  in [1]) for more fair comparison.

Finally, for the bases in Eq.(75), we also construct a filter method based on it. The method is called Fourier-1D method, which shows the relationship of Eq.(75) and the proposed method.

Tables 2 and 3 show the comparison results. From the tables, we can observe that, in the rotation free cases, the RMSE of the Fourier+ method is much larger than the Fourier and the proposed methods, which implies that bandlimiting does hamper the expressive ability of the bases. Besides, we can see that the RMSE of Fourier+ is better than Fourier when the parametrization result is rotated with 45 degree, which implies that bandlimiting helps alleviate the aliasing effect. However, the results of Fourier+ are still worse than the proposed method in this case. More importantly, the RMSE of the Fourier+ and Fourier are almost the same, when representing the object function in 0° and 45° simultaneously. This can be rationally explained by the fact that the discretizations of high frequency bases can be very different in 0° and 45°, which tends to make the high frequency basis hardly be used in representing the object in 0° and 45° simultaneously, and band limiting on Eq. (6) will affect very little. In practice, when we adopt filter parametrization to equivariant convolutions, we actually need to represent a filter under different orientations, and the Fourier+ would not be a good choice in such applications.

Besides, we can observe that the proposed basis set performs better than both Harmonic and Harmonic+ whether rotated or not, which reveals the advantage of the proposed bases over harmonics on this filter parametrization tasks.

#### 4.3 More Equivariance Verifications

We have compared the equivariance errors of CNN, E2-CNN [4] and the proposed F-Conv in the main text. Here, we show more visual results.

The three utilized networks are all with 5 convolutional layers, and each convolutional layer consistently contains

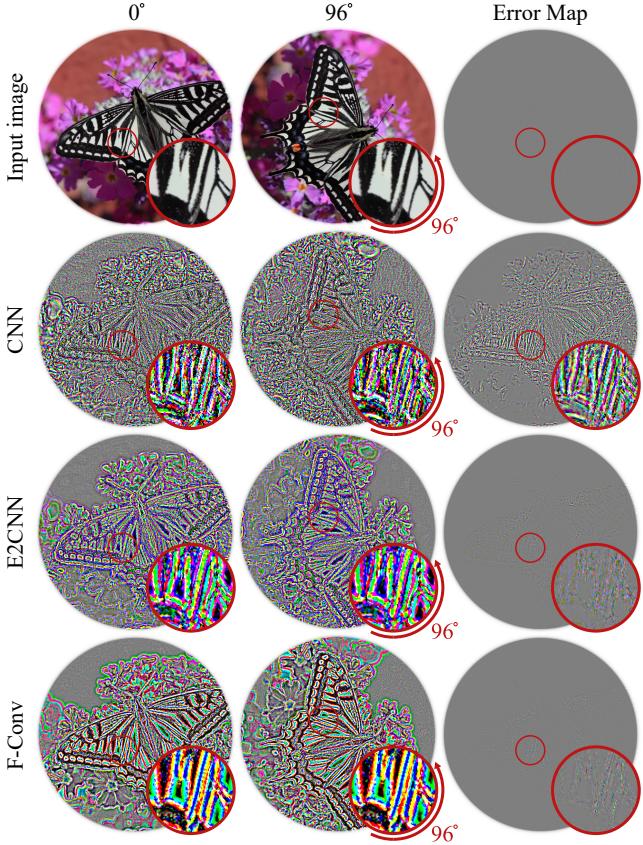


Fig. 8. Performance illustration of CNN, E2-CNN and F-Conv on a sample from the DIV2K dataset. The first column is the results of the rotated free image, the second column is the results the  $96^\circ$  rotated images, and the final column is the error maps of the first column and the  $-96^\circ$  rotated results of the second column.

a convolution, a batch normalization and a ReLU operator. We consider the  $p24$  group for E2CNN and F-Conv methods and set the channel number of all convolution layers in their utilized equivariant networks to be 9 (each channel contains 24 sub-channels correlating to 24 orientations). Besides, we set the channel number of all convolution layers in the utilized CNN to be  $9 \times 24$  so that the three networks take similar computing memory. All three networks are randomly initialized and utilized without training.

Fig. 8 shows the visual results of CNN, E2CNN [4] and F-Conv on an image from Urban100 data set. It is easy to observe that the local features of the outputs of E2-CNN and F-Conv are more stable during rotation, as compared to that of CNN. Besides, their obtained error maps of are clearer than that of CNN.

#### 4.4 Verification on The Function of Output Layer Convolution

In this section, we compare the performance of the proposed output layer convolution and the group pooling operator in low-level version. Note that group pooling is the most commonly used output layer in previous rotation equivariant methods [1], [4], [5].

**Numerical results.** We test the performance of two kinds of rotation equivariant output layers on E2-CNN and the proposed F-Conv method. Specifically, we construct a

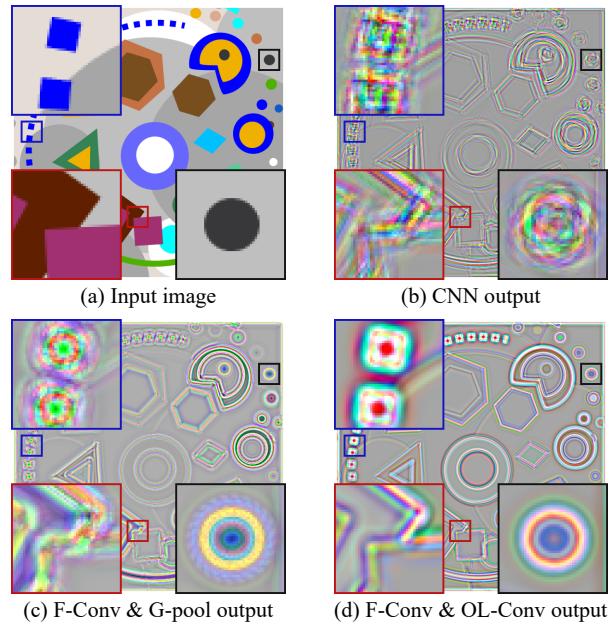


Fig. 9. (a) A typical input cartoon image. (b) Output of randomly initialized CNN and two versions of F-Conv, respectively, where the demarcated areas are zoomed in 5 times for easy observation.

TABLE 4  
The average denoising results of competing networks on 4 image datasets, including Urban100 [6], B100 [7], Set14 [8] and Set5 [9].

Dataset	E2-CNN				F-Conv			
	G-pool		OL-Conv		G-pool		OL-Conv	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Urban100	29.13	0.854	29.25	0.857	29.58	0.864	<b>29.63</b>	<b>0.867</b>
B100	29.06	0.796	29.12	0.798	29.23	0.802	<b>29.28</b>	<b>0.806</b>
Set14	29.06	0.790	29.18	0.792	29.27	0.794	<b>29.36</b>	<b>0.799</b>
Set5	31.20	0.864	31.35	0.867	31.52	0.872	<b>31.56</b>	<b>0.873</b>

lightweight ResNet [10] for our experiments. The network consists of 8 standard residual blocks with kernel size 5, where we consider the  $p8$  group and set 16 multi-channel feature maps for all the correlated intermediate convolution layers (i.e., the total channel number is  $16 \times 8 = 128$ ). For the output layer of the competing methods, we use the following two settings:

- **Group pooling (G-pool).** The group pooling operator is not able to change the channel number. We thus first additionally adopt an intermediate convolution layer group pooling to change the channel number from 16 to 3 (the channel number of RGB image), and then we adopt the group max-pooling operator.
- **Output layer convolution (OL-Conv).** We directly adopt the output layer convolution in this setting, changing channel number from 16 to 3.

For each of the two setting, the rotation equivariance is presented through the entire network.

We adopt the competing networks to a simulated image denoising task. We use 800 training images from the DIV2K dataset [11] to generate the training set. The noisy image is generated by adding Gaussian noise (whose standard deviation is 50) to the clean image. All the networks are trained for 40 epoches with Adam optimizer, with batch

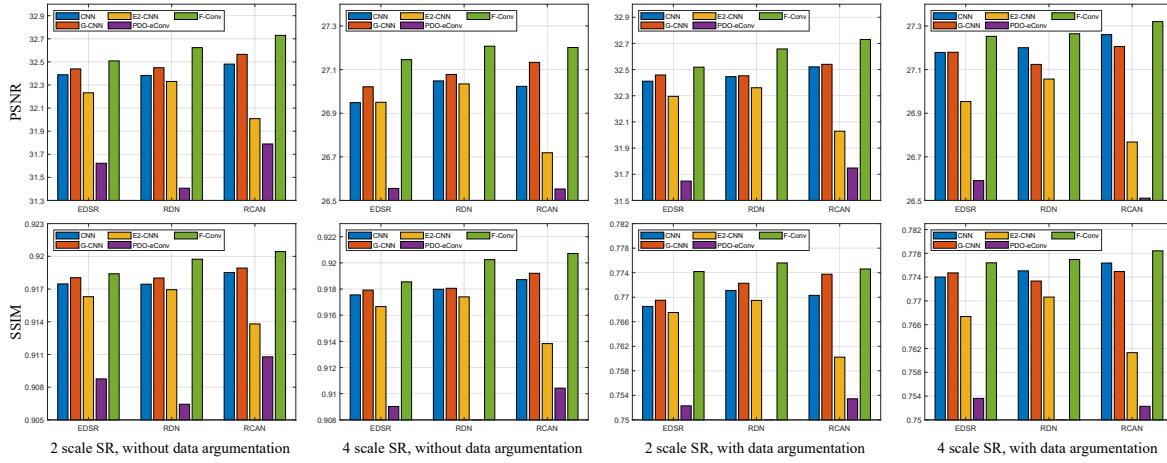


Fig. 10. The Bars of average SR results of all competing methods on the 4 exploited image datasets, including Urban100, B100, Set14 and Set5.

TABLE 5  
Results of all competing methods with similar simple network architecture on MNIST-rot-12k.

Method	Test Error (%)	Params
G-CNN [5]	2.00±0.106	201.7k
E2-CNN [4]	1.23±0.068	44.6k
PDO-eConv [3]	1.79±0.138	60.6k
F-Conv-reduce	1.23±0.091	44.6k
F-Conv-1D	1.27±0.082	100.9k
F-Conv	<b>1.13±0.059</b>	100.9k

size set as 32 and patch size set as 96, respectively. Note that all models are trained without data argumentation. For testing, we use four standard benchmark datasets, including Urban100 [6], B100 [7], Set14 [8] and Set5 [9], which contain 100, 100, 14 and 5 natural images, respectively.

The experimental result are listed in Table 4. From the table, it is easy to observe that the output layer convolution indeed helps consistently improve the performance, as compared with group pooling based output layer.

**Visual results.** Fig. 9 further shows the output of CNN and F-Conv with an cartoon input image, where The network setting is the same as that in Section 5.2 of the main text. For F-Conv, we show its results with setting its output layer as G-pool and OL-Conv, respectively. It is easy to observe that the output of F-Conv with OL-Conv is much shows clearer structure details than F-Conv with G-pool, implying that OL-Conv is much suitable for low-level version tasks.

#### 4.5 More Image Classification Results

**Comparison in simple architectures.** In the main text, we have evaluated the performance of F-Conv with comparison to current equivariant convolutions, under similar networks structure. In this section, we further conduct two more methods for more comprehensive experimental exploration.

Since the proposed method contains more parameters than the state-of-the-art baselines under similar network structure, we conduct a method called F-Conv-reduce, which only utilizes the first 11 principal components of the proposed basis set for filter parametrization and contains exactly the same number of channels and parameters as E2-CNN method. Besides, we have also constructed a version

TABLE 6  
Results of leading board methods and ours on MNIST-rot-12k.

Method	Test Error (%)	Params
H-Net [12]	1.69	0.03M
OR-TIPooling [13]	1.54	≈1M
RotEqNet [14]	1.01	0.10M
PTN-CNN [15]	0.89	0.25M
SFCNN [1]	0.714±0.022	≈3M
E2-CNN [4]	0.682±0.022	≈5M
PDO-eConv [3]	0.709	0.65M
F-Conv-1D	0.684±0.028	3.05M
<b>F-Conv</b>	<b>0.671±0.020</b>	3.05M

of equivariant convolution for the bases in Eq.(75), which is named as F-Conv-1D.

Table 5 shows the results of all competing methods on MNIST-rot-12k. We can see that the F-Conv-reduce achieves comparable performance as E2-CNN in this task. It should be noted that the utilized networks of different methods contains the same channel number, and the difference of these methods in parameter number is mainly attributed to their different basis numbers for representing a filter (E2CNN contains 11 bases, PDO-eConv contains 15 bases, F-Conv contains 25 bases, respectively), which not substantially affect the computing memory and inference efficiency. Therefore, F-Conv actually has achieved better performance than F-Conv-reduce and E2-CNN with only unessentially more computation cost. This verifies that the proposed F-Conv is comparable to the state of art methods on this classification task.

Besides, from the table we can also observe that the performance of F-Conv-1D method is relatively lower than E2-CNN and F-Conv, which implies that the proposed method is a better basis set than that defined in Eq.(75) in this image classification task.

**Comparison with the leading board methods.** We conduct experiments for F-Conv-1D method on a larger network, and compare the performance with leading board methods (the detail experimental setting can refer to Table 5 of the main text). The results are shown in Table 6. It can be observed that the performance of Fourier-1D seems comparable to the E2-CNN method, while is still lower than the proposed F-Conv method.

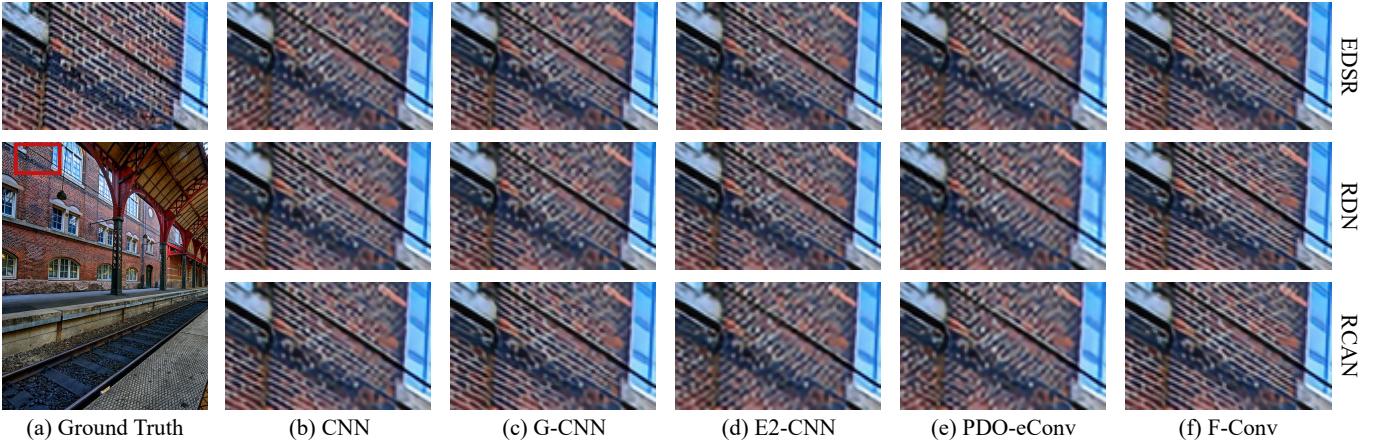


Fig. 11. (a) A sample of high-resolution image from the Urban100 [6] dataset. (b) From upper to lower: the 2 times super-resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as commonly used convolutions, i.e., CNN. (c)-(f) From upper to lower: the super-resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained with data argumentation.

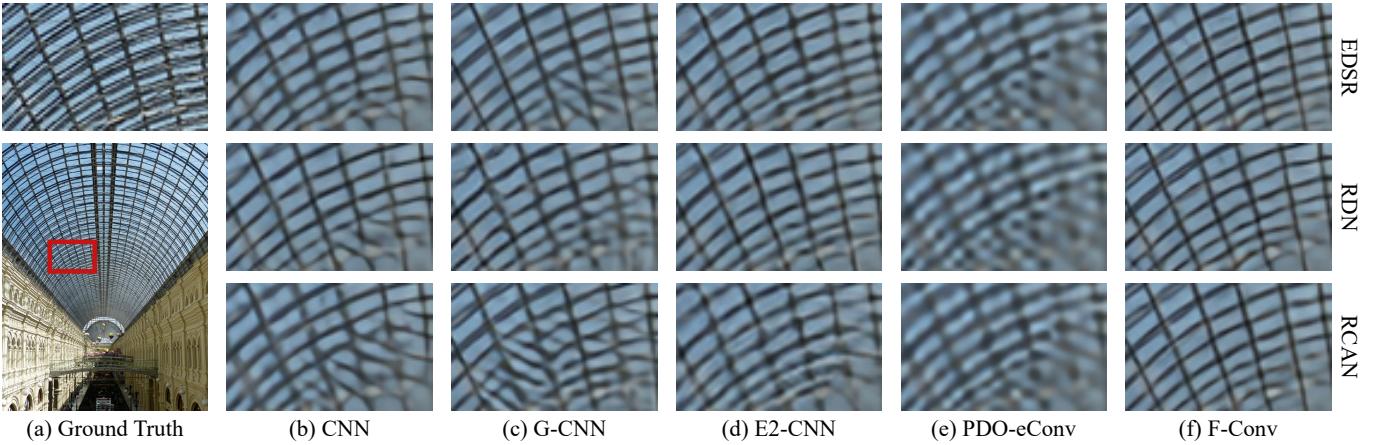


Fig. 12. ((a) A sample of high-resolution image from the Urban100 [6] dataset. (b) From upper to lower: the 4 times super-resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as commonly used convolutions, i.e., CNN. (c)-(f) From upper to lower: the super-resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained without data argumentation.

#### 4.6 Super-resolution Experimental Results

**More results to Section 5.4 of the main text.** In the main text, we have adopted the filter parameterized equivariant convolutions to image super-resolution(SR) in Section 5.4 of the main text. Here, we further show more experimental results.

We have shown the averaging numerical results in Table 7 of the main text. Here we further plot the bars to more intuitively compare the significance of those results in Fig. 10. From the figure, the advantage of the proposed method can be easily observed.

We have shown the visual results on two samples in the Unban100 dataset in the main text, here, we show more visual results on Unban100, B100 and Set 14 datasets. Specifically, Fig. 11 and Fig. 12 show the SR results of the 15 competing methods on 2 samples from the Unban100 dataset, with SR scale set as 2 and 4, trained with and without data augmentation, respectively. Fig. 13 and Fig. 14 show the SR results of the 15 competing methods on 2 samples from the B100 dataset, with SR scale respectively set as 2 and 4, trained with data augmentation. Fig. 15 and

Fig. 16 visually show the SR results of the 15 competing methods on 2 samples from the Set14 dataset, with SR scale respectively set as 2 and 4, trained with data augmentation. From these figures, it is easy to observe that the SR results of filter parametrization based methods like E2-CNN and PDO-eConv are usually over smooth and lack of detail textures. Comparatively, F-Conv based methods performs better in achieving clearer SR image and better avoiding mistakenly restored texture, which are superior to the methods built based on other 4 convolutions.

These results imply that the proposed filter parametrization method should be more suitable for low level computer vision tasks, as compared with the previous Harmonic and PDO based filter parametrization.

**More experiments on image super-resolution.** Finally, we provide more experiments on image super-resolution tasks. The experiment setting is the same as Section 5.4 of the main text. We exploit 3 state-of-the-art networks designed for SR tasks, including EDSR [16], RDN [17] and RCAN [18], for our experiments. The competing equivariant convolution methods include G-CNN [5], E2-CNN [4], PDO-

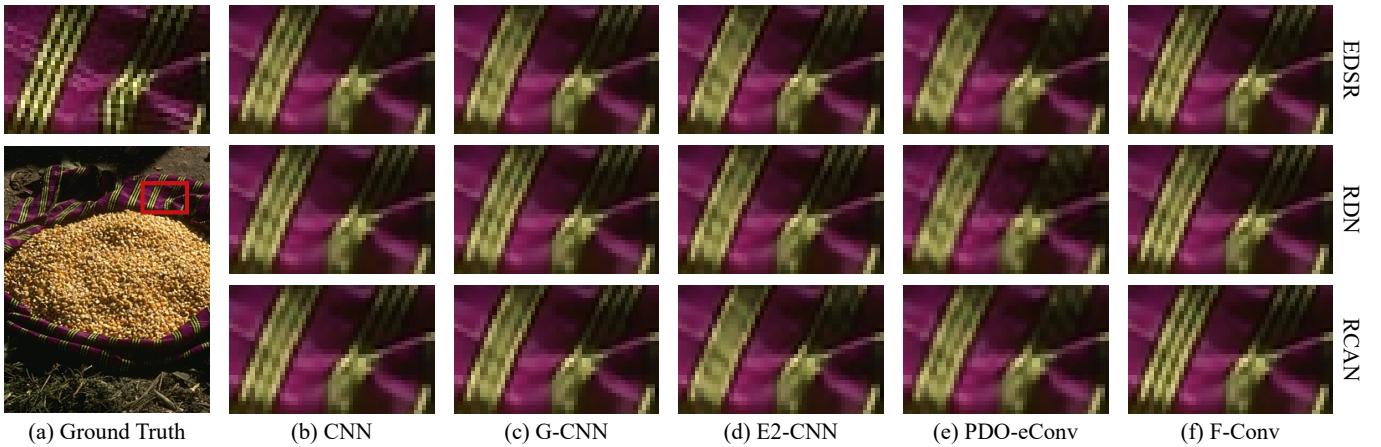


Fig. 13. ((a) A sample of high resolution image from the B100 [7] dataset. (b) From top to bottom are the 4 times super resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are all set as commonly used convolution, i.e., CNN. (c)-(f) From top to bottom are the super resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained with data argumentation.

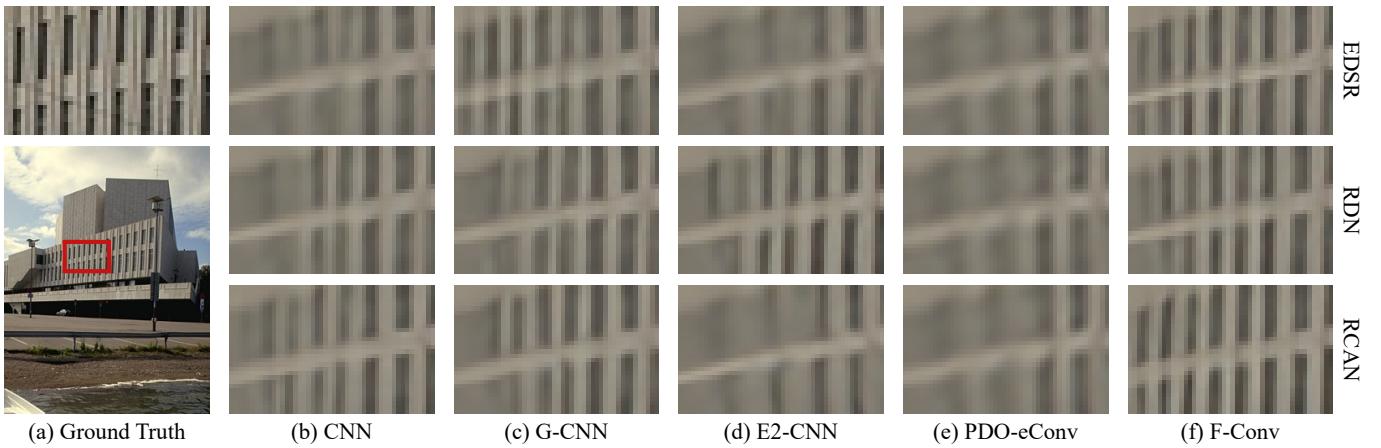


Fig. 14. (a) A typical high resolution image from the B100 [7] dataset. (b) From top to bottom are the 4 times super resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as commonly used convolution, i.e., CNN. (c)-(f) From top to bottom are the super resolution images restored by the EDSR, RDN and RCAN methods, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained with data argumentation.

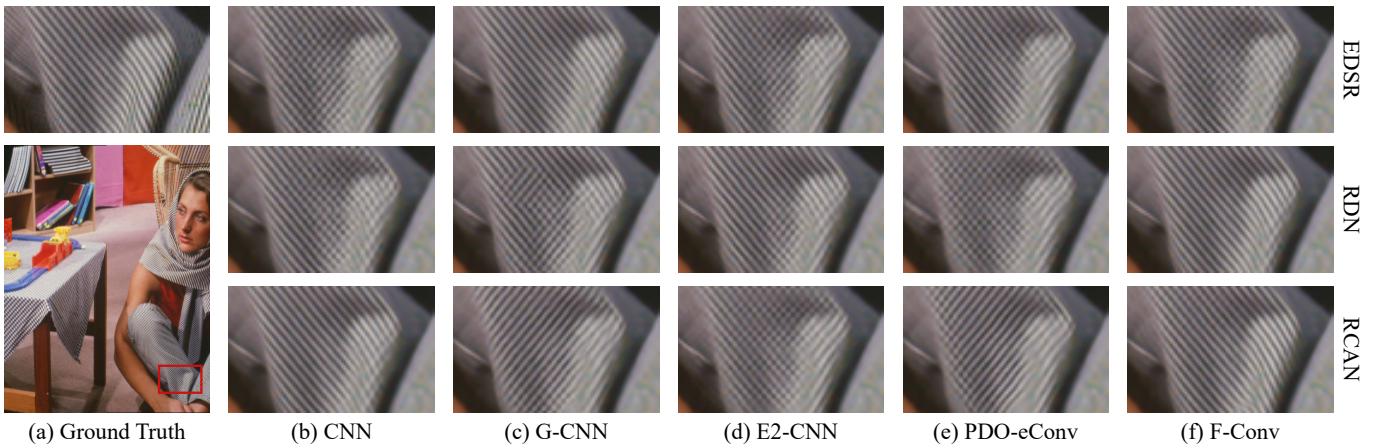


Fig. 15. (a) A typical high resolution image from the Set14 [8] dataset. (b) From top to bottom are the 4 times super resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as commonly used convolution, i.e., CNN. (c)-(f) From top to bottom are the super resolution images restored by the EDSR, RDN and RCAN methods, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained with data argumentation.

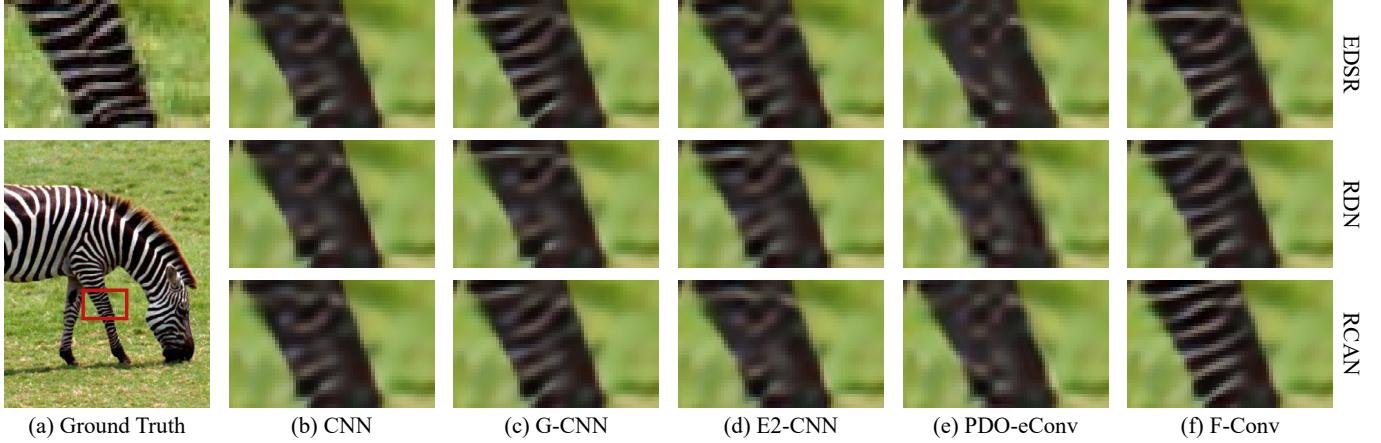


Fig. 16. (a) A typical high resolution image from the Set14 [8] dataset. (b) From top to bottom are the 4 times super resolution images restored by the EDSR, RDN and RCAN methods, respectively, where the convolution operators are set as commonly used convolution, i.e., CNN. (c)-(f) From top to bottom are the super resolution images restored by the EDSR, RDN and RCAN methods, where the convolution operators are set as G-CNN, E2-CNN, PDO-eConv and the proposed F-Conv, respectively. All the involved methods are trained with data argumentation.

TABLE 7

The average testing results of all competing method on 4 typical image datasets, including Urban100 [6], B100 [7], Set14 [8] and Set5 [9], where all models are trained without data argumentation.

Method	$\times 2$						$\times 4$					
	EDSR [16]		RDN [17]		RCAN [18]		EDSR [16]		RDN [17]		RCAN [18]	
	PSNR	SSIM										
Testing data: Urban100												
CNN	32.130	0.9296	32.135	0.9296	32.317	0.9315	25.927	0.7851	26.065	0.7891	26.042	0.7882
G-CNN [5]	32.222	0.9304	32.237	0.9303	32.457	0.9321	26.014	0.7867	26.087	0.7905	26.203	0.7937
E2-CNN [4]	31.867	0.9275	32.051	0.9289	31.516	0.9236	25.901	0.7831	26.023	0.7867	25.544	0.7719
E2-CNN+	31.967	0.9280	32.435	0.9325	32.539	0.9328	26.044	0.7870	26.249	0.7939	26.224	0.7928
PDO-eConv [3]	30.884	0.9156	30.544	0.9118	31.157	0.9189	25.327	0.7598	24.287	0.7196	25.321	0.7615
F-Conv-1D	32.304	0.9309	<b>32.558</b>	<b>0.9336</b>	32.705	0.9340	26.172	0.7908	<b>26.354</b>	<b>0.7977</b>	<b>26.326</b>	<b>0.7964</b>
F-Conv	<b>32.345</b>	<b>0.9312</b>	32.534	0.9332	<b>32.729</b>	<b>0.9345</b>	<b>26.243</b>	<b>0.7948</b>	26.324	0.7968	26.310	0.7952
Testing data: B100												
CNN	32.191	0.9028	32.178	0.9027	32.212	0.9031	27.501	0.7432	27.564	0.7444	27.533	0.7438
G-CNN [5]	32.212	<b>0.9031</b>	32.213	0.9032	32.240	0.9034	27.561	0.7435	27.594	0.7453	27.600	0.7454
E2-CNN [4]	32.129	0.9023	32.158	0.9024	32.027	0.9010	27.529	0.7430	27.576	0.7435	27.415	0.7391
E2-CNN+	32.160	0.9025	32.247	0.9036	32.266	0.9037	27.575	0.7441	27.617	0.7460	27.619	0.7456
PDO-eConv [3]	31.848	0.8983	31.744	0.8972	31.921	0.8993	27.297	0.7344	26.815	0.7202	27.302	0.7352
F-Conv-1D	32.215	<b>0.9031</b>	<b>32.285</b>	<b>0.9039</b>	32.296	0.9040	<b>27.602</b>	<b>0.7448</b>	<b>27.640</b>	<b>0.7466</b>	27.630	<b>0.7464</b>
F-Conv	<b>32.221</b>	<b>0.9031</b>	32.261	0.9037	<b>32.303</b>	<b>0.9041</b>	27.592	0.7454	27.631	0.7462	<b>27.639</b>	0.7458
Testing data: Set14												
CNN	33.637	0.9195	33.593	0.9196	33.585	0.9202	28.452	0.7853	28.546	0.7875	28.539	0.7864
G-CNN [5]	33.630	0.9203	33.660	0.9203	33.688	0.9205	28.576	0.7867	28.648	0.7897	28.621	0.7890
E2-CNN [4]	33.545	0.9198	33.550	0.9190	33.334	0.9179	28.491	0.7857	28.576	0.7866	28.340	0.7809
E2-CNN+	33.572	0.9196	33.708	0.9212	33.843	0.9219	28.578	0.7871	28.642	0.7893	28.628	0.7876
PDO-eConv [3]	33.158	0.9161	32.998	0.9149	33.262	0.9169	28.235	0.7779	27.350	0.7575	28.200	0.7778
F-Conv-1D	33.723	<b>0.9209</b>	33.862	0.9220	33.823	<b>0.9214</b>	<b>28.634</b>	0.7879	28.642	0.7894	28.622	<b>0.7891</b>
F-Conv	<b>33.746</b>	0.9207	<b>33.890</b>	<b>0.9225</b>	<b>33.851</b>	0.9213	28.579	<b>0.7882</b>	<b>28.666</b>	<b>0.7898</b>	<b>28.637</b>	0.7887
Testing data: Set5												
CNN	38.011	0.9621	38.046	0.9621	38.050	0.9622	32.135	0.8972	32.201	0.8981	32.196	0.8981
G-CNN [5]	37.998	0.9619	38.050	0.9622	38.087	0.9624	32.006	0.8961	32.165	0.8982	32.239	0.8985
E2-CNN [4]	37.921	0.9618	37.961	0.9619	37.780	0.9613	32.039	0.8957	32.106	0.8964	31.736	0.8917
E2-CNN+	37.977	0.9620	38.077	0.9624	38.105	0.9623	32.144	0.8971	32.214	0.8982	32.201	0.8975
PDO-eConv [3]	37.581	0.9605	37.443	0.9599	37.675	0.9608	31.589	0.8895	30.292	0.8617	31.584	0.8889
F-Conv-1D	38.041	<b>0.9623</b>	<b>38.124</b>	0.9624	<b>38.151</b>	0.9624	32.189	0.8976	<b>32.346</b>	<b>0.8997</b>	<b>32.280</b>	<b>0.8989</b>
F-Conv	<b>38.061</b>	<b>0.9623</b>	38.122	<b>0.9625</b>	38.149	<b>0.9625</b>	<b>32.259</b>	<b>0.8986</b>	32.317	0.8995	32.265	0.8983

TABLE 8

The average testing results of all competing method on 4 typical image datasets, including Urban100 [6], B100 [7], Set14 [8] and Set5 [9], where all models are trained with data argumentation.

Method	$\times 2$						$\times 4$					
	EDSR [16]		RDN [17]		RCAN [18]		EDSR [16]		RDN [17]		RCAN [18]	
	PSNR	SSIM										
Testing data: Urban100												
CNN	32.170	0.9297	32.230	0.9303	32.359	0.9316	26.250	0.7935	26.303	0.7956	26.395	0.7979
G-CNN [5]	32.269	0.9306	32.237	0.9304	32.396	0.9324	26.264	0.7951	26.147	0.7921	26.316	0.7955
E2-CNN [4]	31.992	0.9283	32.095	0.9294	31.550	0.9238	25.898	0.7829	26.060	0.7885	25.626	0.7737
E2-CNN+	31.986	0.9281	32.492	0.9333	32.572	0.9330	26.055	0.7877	26.337	0.7967	26.354	0.7964
PDO-eConv [3]	30.912	0.9159	29.550	0.8994	31.082	0.9181	25.377	0.7617	24.370	0.7229	25.268	0.7597
F-Conv-1D	<b>32.380</b>	<b>0.9316</b>	32.514	0.9337	<b>32.761</b>	<b>0.9352</b>	26.395	<b>0.7987</b>	26.410	<b>0.7992</b>	26.495	0.8003
F-Conv	32.357	0.9313	<b>32.587</b>	<b>0.9340</b>	32.747	0.9350	<b>26.398</b>	0.7982	<b>26.416</b>	0.7990	<b>26.497</b>	<b>0.8010</b>
Testing data: B100												
CNN	32.201	0.9029	32.215	0.9031	32.240	0.9034	27.641	0.7461	27.638	0.7461	27.667	0.7467
G-CNN [5]	32.209	0.9027	32.217	0.9032	32.243	0.9036	27.629	0.7460	27.629	0.7461	27.638	0.7461
E2-CNN [4]	32.148	0.9024	32.178	0.9028	32.035	0.9009	27.541	0.7429	27.585	0.7442	27.436	0.7396
E2-CNN+	32.161	0.9025	32.260	0.9038	32.281	0.9039	27.585	0.7443	27.648	0.7463	27.642	0.7459
PDO-eConv [3]	31.871	0.8985	31.429	0.8933	31.916	0.8994	27.322	0.7352	26.847	0.7214	27.273	0.7346
F-Conv-1D	32.234	0.9033	32.278	0.9040	32.287	<b>0.9043</b>	<b>27.653</b>	<b>0.7470</b>	<b>27.670</b>	0.7468	27.686	0.7476
F-Conv	<b>32.239</b>	<b>0.9034</b>	<b>32.290</b>	<b>0.9041</b>	<b>32.299</b>	0.9042	27.650	0.7466	27.659	<b>0.7469</b>	<b>27.690</b>	<b>0.7480</b>
Testing data: Set14												
CNN	33.617	0.9198	33.630	0.9204	33.703	0.9202	28.673	0.7894	28.681	0.7904	28.722	0.7904
G-CNN [5]	33.606	0.9197	33.678	0.9204	33.726	0.9210	28.694	0.7901	28.659	0.7892	28.661	0.7893
E2-CNN [4]	33.513	0.9191	33.565	0.9200	33.361	0.9179	28.490	0.7853	28.610	0.7874	28.368	0.7811
E2-CNN+	33.534	0.9192	33.763	0.9218	33.766	0.9215	28.594	0.7874	28.715	0.7905	28.683	0.7895
PDO-eConv [3]	33.169	0.9161	32.511	0.9103	33.178	0.9163	28.241	0.7783	27.422	0.7590	28.171	0.7772
F-Conv-1D	33.656	0.9204	33.766	0.9218	<b>33.825</b>	<b>0.9224</b>	28.700	<b>0.7907</b>	<b>28.747</b>	<b>0.7915</b>	28.717	0.7908
F-Conv	<b>33.698</b>	<b>0.9205</b>	<b>33.832</b>	<b>0.9222</b>	33.762	0.9213	<b>28.715</b>	0.7903	28.706	0.7904	<b>28.743</b>	<b>0.7910</b>
Testing data: Set5												
CNN	38.063	0.9621	38.066	0.9623	38.088	0.9624	32.314	0.8993	32.235	0.8989	32.331	0.8998
G-CNN [5]	38.041	0.9621	38.050	0.9623	38.053	0.9623	32.271	0.8991	32.258	0.8990	32.279	0.8991
E2-CNN [4]	37.912	0.9618	37.964	0.9620	37.789	0.9613	31.999	0.8953	32.085	0.8965	31.761	0.8920
E2-CNN+	37.917	0.9619	38.089	0.9624	38.104	0.9624	32.100	0.8972	32.316	0.8993	32.301	0.8986
PDO-eConv [3]	37.631	0.9606	36.824	0.9566	37.680	0.9608	31.606	0.8899	30.417	0.8641	31.460	0.8874
F-Conv-1D	38.057	<b>0.9622</b>	38.109	0.9624	38.115	<b>0.9625</b>	<b>32.325</b>	<b>0.8996</b>	<b>32.328</b>	<b>0.8997</b>	32.385	0.9000
F-Conv	<b>38.061</b>	<b>0.9622</b>	<b>38.136</b>	<b>0.9625</b>	<b>38.141</b>	0.9624	32.316	0.8993	32.315	<b>0.8997</b>	<b>32.386</b>	<b>0.9001</b>

eConv [3] and the proposed F-Conv. We replace the original convolutions in EDSR, RDN and RCAN with the competing convolutions, respectively.

We have further constructed a version of equivariant convolution based on the Harmonic+ filter parametrization, named E2-CNN+. Since heavy band limiting for current E2-CNN may hamper the performance in low-level image processing tasks, E2-CNN+ would provide a fairer comparison between Harmonic based filter parametrization and the proposed method. We have carefully tuned the parameters of harmonic bases ( $\tau$  and  $K_j$  for  $j = 1, \dots, J$  in [1]) in Harmonic+ filter parametrization for possibly fair comparison. Besides, we also adopt F-Conv-1D (Fourier-1D filter parametrization based equivariant convolution) in this experiment to test the performance of Fourier-1D filter parametrization.

Table 7 and Table 8 show the results with respect to the 4 utilized datasets, without and with data argumentation, respectively. From the tables, we can observe that E2-CNN+ achieves better performance than E2-CNN, however, the performance is still relatively worse than F-Conv method. These results imply that better expression ability of filter parametrization would lead to better performance in

low-level tasks. Besides, the F-Conv-1D and F-Conv methods achieve comparable performance, while both outperform the other competing methods. These results show the advantage of the Fourier series expansion based filter parametrization in conducting equivariant convolutions for low-level image processing tasks.

## REFERENCES

- [1] Maurice Weiler, Fred A Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 849–858, 2018.
- [2] Soo-Chang Pei and Min-Hung Yeh. Two dimensional discrete fractional fourier transform. *Signal Processing*, 67(1):99–108, 1998.
- [3] Zhengyang Shen, Lingshen He, Zhouchen Lin, and Jinwen Ma. Pdo-econvs: Partial differential operator based equivariant convolutions. In *International Conference on Machine Learning*, pages 8697–8706. PMLR, 2020.
- [4] Maurice Weiler and Gabriele Cesa. General  $e(2)$ -equivariant steerable cnns. 2019.
- [5] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 2990–2999. PMLR, 2016.
- [6] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015.

- [7] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001.
- [8] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010.
- [9] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [11] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017.
- [12] Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [13] Yanzhao Zhou, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Oriented response networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [14] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [15] Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis. Polar transformer networks. *arXiv preprint arXiv:1709.01889*, 2017.
- [16] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.
- [17] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018.
- [18] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018.