Fourier Image Transformer

Tim-Oliver Buchholz
FMI for Biomedical Research
Basel, Switzerland
tim-oliver.buchholz@fmi.ch

Florian Jug
Fondazione Human Technopole
Milan, Italy
florian.jug@fht.org

Abstract

Transformer architectures show spectacular performance on NLP tasks and have recently also been used for tasks such as image completion or image classification. Here we propose to use a sequential image representation, where each prefix of the complete sequence describes the whole image at reduced resolution. Using such Fourier Domain Encodings (FDEs), an auto-regressive image completion task is equivalent to predicting a higher resolution output given a low-resolution input. Additionally, we show that an encoder-decoder setup can be used to query arbitrary Fourier coefficients given a set of Fourier domain observations. We demonstrate the practicality of this approach in the context of computed tomography (CT) image reconstruction. In summary, we show that Fourier Image Transformer (FIT) can be used to solve relevant image analysis tasks in Fourier space, a domain inherently inaccessible to convolutional architectures.

1. Introduction

Transformer architectures are currently setting new standards on virtually all natural language processing (NLP) tasks [7, 21]. But also in other domains, transformers can find application [15]. Vision Transformers (ViT) [8], for example, show SOTA results on image classification tasks. In the future we expect an increasing number of problems to be solved with transformers and here we also contribute such applications.

The key novelty of transformers is their self-attention mechanism [26], allowing them to learn and utilize long ranging dependencies in data. Recently, this mechanism is applied to longer and longer input sequences of words or other elements, *e.g.* pixels [5].

Initially, we wondered if pixel sequences are the only valid image representation to train *auto-regressive transformer* models in the spirit of [5]. In particular, we want to use a representation where each prefix of such a descriptive sequence encodes the full image at lower resolution.
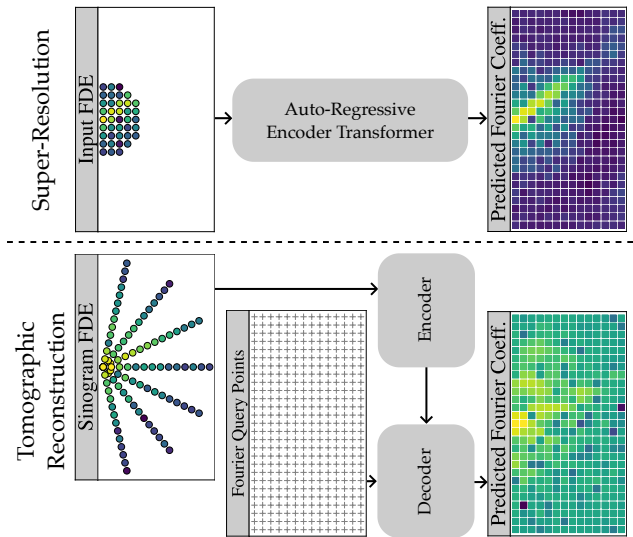

Figure 1. We present Fourier Image Transformers (FITs), realizations of Fast-Transformers [12], that operate on images via a novel sequential image representation we call Fourier Domain Encoding (FDE). We demonstrate the utility of FITs on two tasks. First, an image super-resolution task *(top row)*, where the low frequencies of an image are used to predict missing high frequencies, corresponding to a higher resolution image. The second task is tomographic reconstruction *(bottom row)*, where we provide a sparse sampled Fourier space as input to an encoder-decoder FIT and predict missing Fourier coefficients. Note that the real images can be obtained from both predictions by taking the inverse Fourier transform.

Hence, we introduce *Fourier Domain Encodings* (FDEs), which do have this desired property and, as we show, can successfully be used to train auto-regressive Fourier Image Transformers ("FIT: SRes") for super-resolution tasks (see Figures 1 and 2).

Additionally, we show how an *encoder-decoder* based Fourier Image Transformer ("FIT: TRec") can be trained on a set of Fourier measurements and then used to query arbitrary Fourier coefficients, which we use to improve sparse-

view computed tomography (CT) image restoration[1]. We demonstrate this by providing a given set of projection Fourier coefficients to our encoder-decoder setup and use it to predict Fourier coefficients at arbitrary query points. This allows us to predict a dense, grid-sampled discrete Fourier spectrum of a high quality CT reconstruction (see Figures 1 and 3).

In Section 2 we review some of the related transformer and tomography literature. In Section 3 we introduce our novel Fourier Domain Encoding (FDE) and training strategies for auto-regressive and encoder-decoder transformer models. In Section 4 we present our experiments and results on multiple datasets.

2. Related Work

Transformer architectures are revolutionizing neural language processing (NLP), replacing recurrent neural networks (RNNs) and long short-term memory (LSTM) architectures in virtually all NLP tasks [7, 21]. The success of transformers in NLP has naturally raised the question if computer vision tasks might as well benefit from transformer-like attention. Recent work on image classification [8, 23] and pixel-by-pixel image generation/completion [5, 12, 20] were among the first to successfully demonstrate the applicability of transformers in the image-domain. In [20], for example, the first n pixels of the flattened input image are used to condition a generative transformer setup that then predicts the remaining image in an auto-regressive manner.

Concurrently with our own work, Lee-Thorp *et al.* [16] have proposed an interesting idea to combine Fourier-space image information with Transformer-based attention. While their approach is inspiring and leads to much improved results on a number of tasks on data other than images, our goals of super-resolution and tomographic reconstruction are of different nature and bear no overlap with their investigations.

2.1. Attention is all you need

Vaswani *et al.* [26] were the first to introduce transformers. More specifically, they introduced an encoder-decoder structure, where the encoder maps an input encoding $\mathbf{x} \in \mathbb{R}^{N \times F}$ into a continuous latent space $\mathbf{z} \in \mathbb{R}^{N \times F}$, with N corresponding to the number of input tokens and F representing the feature dimensionality per token. This latent space embedding $\mathbf{z}$ is then given to the decoder, which generates an M long output sequence $\mathbf{y} \in \mathbb{R}^{M \times F}$ iteratively, element by element. This auto-regressive decoding scheme means that the decoder generated the i-th output token while not only observing $\mathbf{z}$, but also all $i-1$ output tokens generated previously.

---

[1]Sparse-view CT image restoration typically suffers from missing high frequency Fourier coefficients, leading to unwanted image artefacts.

More formally, a transformer is a function $T : \mathbb{R}^{N \times F} \to \mathbb{R}^{N \times F}$, represented by L transformer layers

$$T_l(\mathbf{x}) = f_l(A_l(\mathbf{x}) + \mathbf{x}), \tag{1}$$

with A_l denoting a self-attention module and f_l being a simple feed forward network.

In the self-attention module, the input $\mathbf{x}$ is mapped to queries $Q = \mathbf{x}W_Q$, keys $K = \mathbf{x}W_K$ and values $V = \mathbf{x}W_V$ by matrix multiplication with learned matrices $W_Q \in \mathbb{R}^{F \times D}, W_K \in \mathbb{R}^{F \times D}$ and $W_V \in \mathbb{R}^{F \times F}$. The self-attention output is then computed by

$$A_l(\mathbf{x}) = \text{softmax}\left(\frac{QK^T}{\sqrt{D}}\right)V, \tag{2}$$

with the softmax-function being applied per row. Intuitively, the softmax-normalized similarity between computed keys and queries is used to obtain the weighted sum over the values.

Typically, instead of a single self-attention module, multi-head attention is being used. If that is the case, a transformer layer T_l learns multiple W_Q, W_K and W_V, allowing the layer to simultaneously perform multiple attention-based computations [26].

Since transformers do not explicitly encode the relative position between input tokens, positional encodings are required whenever specific input topologies need to be made accessible to the transformer. In [26], a useful 1D positional encoding scheme was proposed. Later, Wang *et al.* [28] generalized this scheme to 2D topologies. In our own work, we have adopted this encoding scheme, but use it not only to encode integer pixel-grid locations, but arbitrary coordinates.

While the advantage of transcending beyond a CNN's localized receptive field by introducing global attention proves beneficial for many learning tasks, the big downside is the computational cost that comes with it. Due to the required matrix multiplication QK^T, the self-attention on an input sequence of length N requires $\mathcal{O}(N^2)$ memory and time. Since image-based applications have to deal with very long input sequences, the use of efficient transformer implementations [2, 13, 27] is essential. Hence, we use fast-transformers, an efficient approximation of softmax self-attention, introduced by Katharopoulos *et al.* [12].

2.2. Tomographic Image Reconstruction

In computed tomography (CT), the radon transform [11, 22] of a 2D section is acquired by rotating a 1D detector array around it, acquiring a series of density measurements at projection angles α_i. Such a series of acquisitions is often depicted as a 2D image, a so called sinogram, in which each pixel column corresponds to one such 1D measurement. A widely used tomographic reconstruction method is filtered
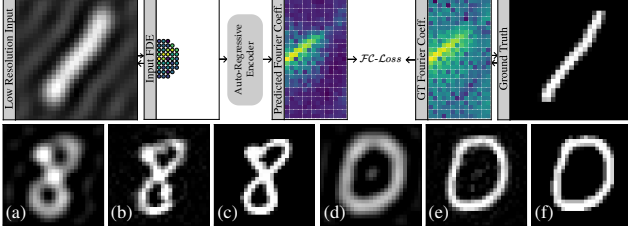
Figure 2. **FIT for super-resolution.** Low-resolution input images are first transformed into Fourier space and then unrolled into an FDE sequence, as described in Section 3.1. This FDE sequence can now be fed to a FIT, that, conditioned on this input, extends the FDE sequence to represent a higher resolution image. This setup is trained using an $\mathcal{FC}\text{-}\mathcal{Loss}$ that enforces consistency between predicted and ground truth Fourier coefficients. During inference, the FIT is conditioned on the first 39 entries of the FDE, corresponding to *(a,d)* 3× Fourier binned input images. Panels *(b,e)* show the inverse Fourier transform of the predicted output, and panels *(c,f)* depict the corresponding ground truth.
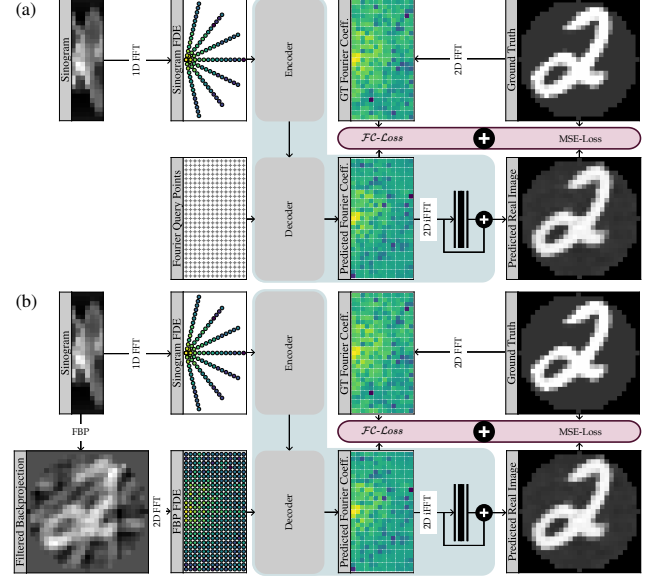

Figure 3. **FIT for computed tomography.** In 2D computed tomography, 1D projections of an imaged sample (*i.e.* the columns of a sinogram) are back-transformed into a 2D image. We propose two encoder-decoder based Fourier Image Transformer setups for tomographic reconstruction. For "FIT: TRec", subfigure (*a*), the sinogram is converted into our proposed FDE which serves as input to the encoder. The decoder predicts a dense Fourier spectrum from the latent encoding and provided Fourier query points. In order to reduce high frequency fluctuations in this result, we introduce a shallow conv-block after the iFFT (shown in black). The setup is trained with the $\mathcal{FC}\text{-}\mathcal{Loss}$, see Section 3.2, and a conventional MSE-loss between prediction and ground truth. For "FIT: TRec + FBP", sub-figure (*b*), we enrich the Fourier query points with encoded information from the filtered backprojection (FBP) [11, 24].

backprojection (FBP) [11, 24] (see Figure 3). FBP is based on the Fourier slice theorem, which states that the Fourier coefficients of each 1D projection at a given angle α_i coincide with the 2D Fourier coefficients that lie on the line that crosses through the DC component of the Fourier space representation at angle α_i [3]. Hence, we can compute the 1D Fourier transformation of the 1D projection measurements and arrange them according to their projection angle α_i in 2D Fourier space, take the inverse Fourier transformation and thereby obtain the reconstructed 2D image.

In practice, it is desirable to limit the number of projections in order to reduce overall acquisition times and total sample exposure. However, having fewer measurements leads wedges of missing Fourier measurements, which then lead to reconstruction artefacts. More specifically, such missing wedges lead to radial striping artefacts in reconstructed images, best observed around image locations that are subject to large intensity gradients (see *e.g.* Figure 3).

A plethora of methods to reduce these artefacts was proposed [1, 4, 9, 10], but none of them is operating in Fourier space. Below, we will introduce Fourier Domain Encodings (FDEs) which is based on the Fourier transformation of an image. Then we train a transformer to fill in all unobserved Fourier coefficients such that missing-wedge artefacts will be reduced or, ideally, avoided.

3. Methods

3.1. Fourier Domain Encodings (FDEs)

To compute the Fourier Domain Encoding (FDE) for an image x, we first take the discrete Fourier transform (DFT), $X = \mathcal{F}(x)$, giving us the complex valued Fourier spectrum X. The DC component of X is in its center-most

location. Concentric rings of Fourier coefficients around the DC component are called Fourier rings. More central Fourier rings contain lower frequencies, coefficients further away from the DC component higher ones. Since we started with a real-valued image x, we know that the Fourier spectrum is radially symmetric, which allows us to drop half of X, resulting in X_h. A lower resolution real image x_r of the original image x can be obtained by masking all Fourier coefficients above a given radius r and computing the inverse Fourier transform of the result. Effectively the real image resolution depends entirely on the value of r, see Figure 2 for an example.

For the Fourier Domain Encoding (FDE) we convert the complex Fourier coefficient rings into a 1D sequence starting form the DC component followed by an unrolling of the (half) Fourier rings from X_h

$$\mathbf{S} = \text{unroll}(X_h) = [c_1, c_2, \ldots, c_N]^T, \qquad (3)$$

where c_i – the words of our input sequence – correspond to

the complex Fourier coefficients. The sequence $\mathbf{S}$ has the interesting property that any sub-sequence $\mathbf{S}_j$ starting at c_1 and ending at $c_j, j \in [2, N]$ encodes the whole real image at lower resolutions.

In order to proceed, we convert the complex Fourier coefficients c_i into normalized amplitudes

$$a_i = \frac{2(|c_i| - a_{max})}{a_{max} - a_{min}} - 1 \qquad (4)$$

and phases

$$\phi_i = \frac{\angle(c_i)}{\pi}, \qquad (5)$$

where a_{min} and a_{max} are minimum and maximum amplitudes computed over all training images and the function $\angle$ returns the phase of a given Fourier coefficient. Hence, the complex sequence $\mathbf{S}$ can now be described by the normalized real-valued matrix

$$\mathbf{C} = \begin{bmatrix} a_1 & \cdots & a_N \\ \phi_1 & \cdots & \phi_N \end{bmatrix}^T \qquad (6)$$

with $\mathbf{C} \in \mathbb{R}^{N \times 2}$.

Our final goal is to transform each word (a_i, ϕ_i) into an F-dimensional vector. To this end we feed $\mathbf{C}$ through a single trainable linear layer that increases the feature dimensionality from 2 to $\frac{F}{2}$, to which we concatenate an $\frac{F}{2}$-dimensional 2D positional encoding[2] that represents the original polar coordinates of the Fourier coefficient in the original 2D Fourier spectrum $\mathbf{X_h}$. The final FDE image sequence is therefore $\mathbf{E} \in \mathbb{R}^{N \times F}$. The sub-sequence property of $\mathbf{S}_j$ holds for $\mathbf{E}_j$ as well.

Predicted output words $\mathbf{Z} = [z_1, \ldots, z_k]$, with $z_i \in \mathbb{R}^F$, are fed through two linear layers that back-transform the F-dimensional encoding of z_i into predicted amplitudes and phases $\hat{c}_i = (\hat{a}_i, \hat{\phi}_i)$, respectively. The output of the phase-predicting layer is additionally passed through the $tanh$-activation function to ensure that all phases are in $[-1, 1]$.

3.2. Fourier Coefficient Loss

We train our Fourier Image Transformers (FITs) with a loss function consisting of two terms, (i) the amplitude loss

$$\mathcal{L}_{amp}(\hat{a}_i, a_i) = 1 + (\hat{a}_i - a_i)^2, \qquad (7)$$

computed between the predicted amplitudes $\hat{a}_i$ and the target amplitudes a_i, and (ii) the phase loss

$$\mathcal{L}_{\angle}(\hat{\phi}_i, \phi_i) = 2 - cos(\hat{\phi}_i - \phi_i), \qquad (8)$$

with $\hat{\phi}_i$ the predicted phase and ϕ_i the corresponding target phase.

---

[2]We slightly adapted the 2D positional encoding of [28] to accept arbitrary (non integer) coordinates.

The final Fourier coefficient loss $\mathcal{L}_{\mathcal{FC}}$ is the multiplicative combination of both individual losses, given by

$$\mathcal{L}_{\mathcal{FC}}(\hat{\mathbf{C}}, \mathbf{C}) = \frac{1}{N} \sum_{i=0}^{N} \mathcal{L}_{amp}(\hat{a}_i, a_i) \cdot \mathcal{L}_{\angle}(\hat{\phi}_i, \phi_i). \qquad (9)$$

3.3. FIT for Super-Resolution

The sub-sequence property of the Fourier domain encoding (FDE) $\mathbf{E}$, where each $\mathbf{E}_j$ encodes a lower resolution image $\boldsymbol{x}_j$ enables us to train an auto-regressive Fourier Image Transformer for super-resolution ("FIT: SRES").

This FIT for super-resolution ("FIT: SRes") gets an FDE sequence $\boldsymbol{E}_{N-1} = [e_1, \ldots, e_{N-1}]$ as input and is trained w.r.t. the correct target sequence $\boldsymbol{Z} = [z_2, \ldots, z_N]$, using the previously introduced Fourier coefficient loss $\mathcal{L}_{\mathcal{FC}}$, computed on the back-transformed predicted amplitude and phase values $\hat{\boldsymbol{C}} = [\hat{c}_2, \ldots, \hat{c}_N]$, where $\hat{c}_i = (\hat{a}_i, \hat{\phi}_i)$, as explained in Section 3.1.

Once the transformer is trained, a sub-sequence $\boldsymbol{E}_j = [e_1, \ldots, e_j], j \in [2, N]$ is used to condition the transformer, which is then used in iterations to auto-regressively predict the missing part of the complete sequence $\boldsymbol{E} = [e_1, \ldots, e_N]$, i.e. filling in predicted high-frequency information not contained in $\boldsymbol{E}_j$ (see Figure 2).

Note that the proposed super-resolution setup operates exclusively on Fourier domain encoded data. All final prediction images $\hat{\boldsymbol{x}}$ are generated by computing the inverse Fourier transform on predictions $\hat{\boldsymbol{C}}$, which are rearranged (rolled) into $\hat{\boldsymbol{X}}_{\boldsymbol{h}}$ and completed to a full predicted Fourier spectrum $\hat{\boldsymbol{X}}$, i.e. $\hat{\boldsymbol{x}} = \mathcal{F}^{-1}(\text{roll}(\hat{\boldsymbol{C}}))$.

3.4. FIT for Tomography

Our Fourier Image Transformer setup for tomographic reconstruction ("FIT: TRec") is based on an encoder-decoder transformer architecture as shown in Figure 3.

As input to the encoder we use the Fourier Domain Encoding (FDE) of a raw sinogram $\boldsymbol{s}$. As described above, $\boldsymbol{s}$ consists of P pixel columns $[s_1, \ldots, s_P]$ of 1D projections of $\boldsymbol{x}$ at angles $[\alpha_1, \ldots, \alpha_P]$. The Fourier slice theorem states, see also Section 2.2, that the discrete 1D Fourier coefficients $\boldsymbol{C}_i = \mathcal{F}(s_i)$ coincide with the values of the 1D slice at angle α_i through the 2D Fourier spectrum $\mathcal{F}(\boldsymbol{x})$. To assemble the full FDE of a sinogram we need to combine all $\boldsymbol{C}_i$ with the adequate positional encoding (using polar coordinates) of all Fourier coefficients, as dictated by the Fourier slice theorem and sketched in Figure 3.

Hence, the encoder creates a latent space representation $\boldsymbol{Z}$ that encodes the full input sinogram $\boldsymbol{s}$. This latent space encoding is then given as input to the decoder. The decoder is then used to predict all Fourier coefficients $\hat{\boldsymbol{C}}$, such that the predicted reconstruction $\hat{\boldsymbol{x}}$ of $\boldsymbol{x}$ can be computed by $\hat{\boldsymbol{x}} = \mathcal{F}^{-1}(\text{roll}(\hat{\boldsymbol{C}}))$, where roll arranges the 1D sequence back
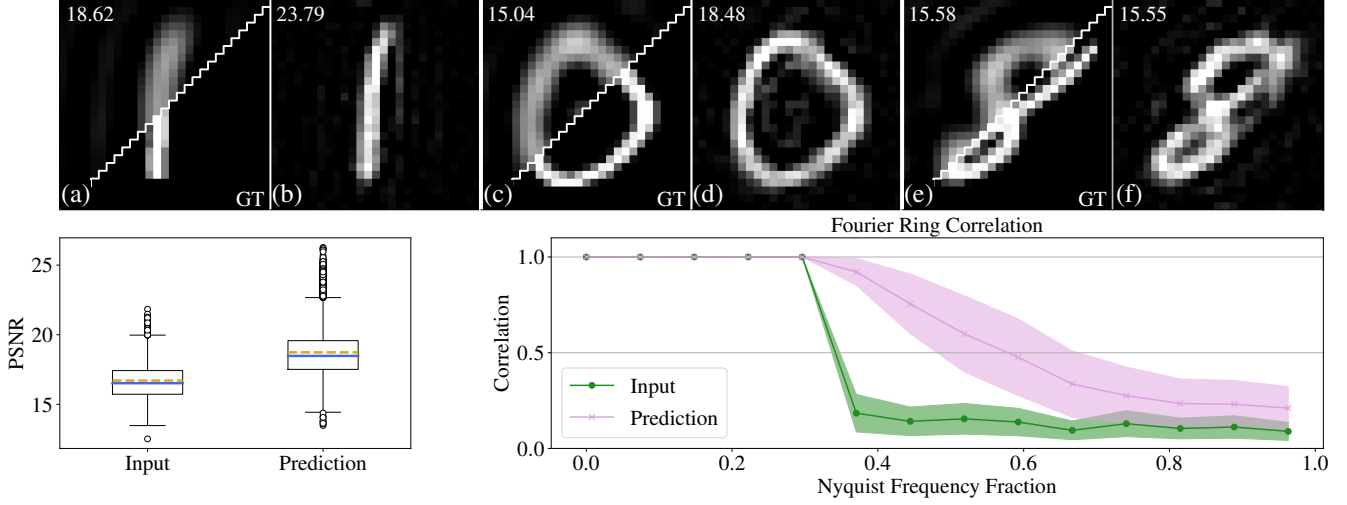
Figure 4. **Super-resolution results on MNIST.** The top left triangles of *(a,c,e)* show $3\times$ binned (low-res) MNIST inputs. Conditioned by these inputs, our trained FIT auto-regressively generates results as shown in *(b,d,f)*. Inputs and predictions are labeled with the peak signal-to-noise ratio (PSNR) values computed w.r.t. ground truth images, shown in the lower-right half of *(a,c,e)*. The examples in *(a-f)* correspond to the 98th, 50th and 2nd percentile in terms of obtained PSNR over all MNIST prediction results. Box-plots show the distribution of PSNR values of Fourier binned inputs and predicted outputs, respectively (mean in dashed gold and median in solid blue). The Fourier ring correlation plot shows how predicted Fourier coefficients are improving w.r.t. true ground truth coefficients. Shaded areas correspond to $+/-1$ standard deviation. The correlation for the first 5 Fourier shells is 1 because these shells have been used as inputs to the FIT.

into a discrete 2D Fourier spectrum. We call this setup "FIT: TRec".

Additionally, we propose a variation of this procedure, called "FIT: TRec + FBP", where the decoder not only receives the latent space encoding $\boldsymbol{Z}$, but also FDEs of the Fourier coefficients $\boldsymbol{C}_{\text{FBP}} = \mathcal{F}(\text{FBP}(\boldsymbol{s}))$, where FBP denotes the function computing the filtered backprojection of a sinogram (see Figure 3).

Note that the implementation of "FIT: TRec" coincides with "FIT: TRec + FBP", with FBP being replaced by a function ZERO which returns 0 for all inputs.

We train "FIT: TRec" and "FIT: TRec + FBP" using the $\mathcal{L}_{\mathcal{FC}}$-loss of Eq.9. Additionally, we introduced a residual convolution block consisting of two convolutional layers (3×3 followed by 1×1) with $d_{\text{conv}} = 8$ intermediate feature channels. This conv-block (conv) receives the inverse Fourier transform of the predicted Fourier coefficients $\hat{\boldsymbol{x}} = \mathcal{F}^{-1}(\text{roll}(\hat{\boldsymbol{C}}))$ as input and is trained using the MSE-loss between the predicted real-space image conv$(\hat{\boldsymbol{x}})$ and the known ground truth image $\boldsymbol{x}$. Hence, the full loss is the sum over $\mathcal{L}_{\mathcal{FC}}$ and the MSE-loss.

In order to speed up training, we start by feeding only a low-resolution subset of all Fourier coefficients $\boldsymbol{C}_i = \mathcal{F}(s_i)$ (and $\boldsymbol{C}_{\text{FBP}}$), and successively increase this subset over training until the full sets are used. This forces the FIT to first learn good low resolution features and later learn to add suitable high resolution predictions.

4. Experiments and Results

Here we describe the experiments we conduct with the previously described super-resolution ("FIT: SRes") and tomographic reconstruction ("FIT: TRec" and "FIT: TRec + FBP") setups. Note, all code used to create results reported in this manuscript is available on GitHub[3].

4.1. Data and Metrics for Super-Resolution

MNIST [14]: Original images are cropped to 27×27 pixels with the default PyTorch train-test split (in $60'000$ and $10'000$ images, respectively). The train images are further split into $55'000$ samples for training and $5'000$ validation images.

CelebA $\mathbf{128 \times 128}$ *[19]:* Original images are converted to gray scale and downscaled to 63×63 pixels. The images are randomly split into $20'000$, $5'000$ and $5'000$ training, validation and test samples, respectively.

We evaluate all results using (i) Fourier Ring Correlation (FRC) [25] in Fourier space and (ii) the peak signal-to-noise ratio (PSNR) in image space.

4.2. Data and Metrics for Computed Tomography

Data Preparation for Tomographic Reconstruction: As described in Section 2.2, tomographic image reconstruction

---

[3] https : / / github . com / juglab / FourierImageTransformer

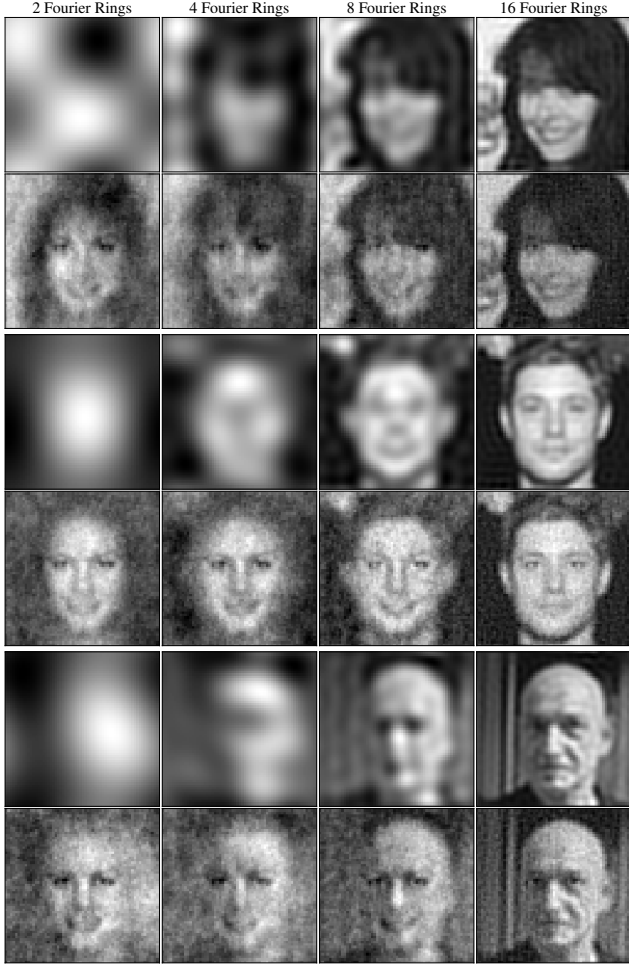| 2 Fourier Rings | 4 Fourier Rings | 8 Fourier Rings | 16 Fourier Rings |

Figure 5. **Super-resolution results on CelebA.** On 3 input images (rows 1+2, 3+4, and 5+6) we show predictions of a trained super-resolution FIT conditioned on 2, 4, 8 and 16 Fourier rings (columns). Upper rows show the iFFT of the input FDEs used to condition the FIT, lower rows depict the iFFT of the predicted results.

in 2D operates on a number of 1D projections of a given true object x. Our tomographic simulation process is based on the work by Leuschner *et al.* [17]. Furthermore, we chose the detector length to be equal to the width of our chosen object (*i.e.* ground truth image) we apply our synthetic tomography pipeline on. To avoid spurious contributions to individual projections, we need to set all pixel intensities outside the largest image-centered circle to 0 (hence, we see only circular images in Figures 3, and 6).

MNIST [14]: Data is split and preprocessed as described in Section 4.1. Additionally, for visualization purposes, we min-clip all pixel intensities within the before-mentioned largest image-centered circle to 50. Finally, we use this data to compute $P = 7$ equally spaced projections which we assemble in sinograms $s^j_{\text{MNIST}} = [s^{j,1}_{\text{MNIST}}, \ldots, s^{j,P}_{\text{MNIST}}]$.

Kanji [6]: Data is randomly split into $50'000$ train, $5'000$ validation and $5'000$ test samples and all images are cropped to 63×63 pixels, which are otherwise processed as described for MNIST. Finally, we use this data to compute $P = 33$ equally spaced projections which we assemble in sinograms $s^j_{\text{Kanji}} = [s^{j,1}_{\text{Kanji}}, \ldots, s^{j,P}_{\text{Kanji}}]$.

LoDoPaB [17]: The original train- and validation-data is first reduced to $4'000$ and 400 randomly chosen images respectively and for testing all $3'553$ images are used. All selected images are downscaled to 111×111 pixels and we compute $P = 33$ equally spaced projections like for the Kanji data.

All tomographic reconstruction experiments with "FIT: TRec" and "FIT: TRec + FBP" and the FBP baseline are evaluated using peak signal-to-noise ratio (PSNR) w.r.t. available ground truth.

4.3. Training Setup for Super-Resolution

We use a $F = 256$ dimensional FDE, with the positional encoding being based on polar coordinates, *i.e.* Fourier coefficients of same frequency have the same radius. The FDE is passed to a *causal-linear* transformer [12] with 8 layers, 8 self-attention heads, a query and value dimensionality of 32, dropout of 0.1, attention dropout of 0.1, and a dimensionality of the feed-forward network of 1024.

This setup is trained auto-regressively, *i.e.* with a triangular attention mask. We use the rectified Adam optimizer (RAdam) [18] with an initial learning rate of 0.0001 and weight decay of 0.01 for 100 epochs. The batch size is 32. We half the learning rate on plateauing validation loss.

4.4. Results for Super-Resolution Tasks

Quantitative results for all conducted super-resolution experiments on MNIST data are shown in Figure 4, where we show (i) $3\times$ binned low-res MNIST input images corresponding to $\boldsymbol{E}_{pre} = [c_1, \ldots, c_{39}]$, also sketched in Figure 2, (ii) corresponding ground truth MNIST images, (iii) the predictions of the "FIT: SRes" network trained on the MNIST data as described in Section 3.3, (iv) two box-plots showing the distribution of PSNR values computed between the ground truth images and the downscaled inputs and predicted outputs, respectively, and (v) Fourier ring correlation plots showing the correlation between our predicted Fourier coefficients and the corresponding ground truth.

In Figure 5, we show 3 sequences of super-resolution results obtained with a FIT trained on the CelebA data. For each image, we conditioned the trained transformer on 2, 4, 8, and 16 Fourier rings, respectively. This corresponds to low-resolution images subject to $16\times$, $8\times$, $4\times$, and $2\times$ binning in Fourier space, respectively.

Dataset	Method	PSNR
MNIST [14]	Baseline: FBP	17.87
	FIT: TRec + FBP (Ours)	27.85
	FIT: TRec (Ours)	**27.90**
	Ablation: Only FBP	26.89
	Ablation: Only Conv-Block	22.53
Kanji [6]	Baseline: FBP	22.06
	FIT: TRec + FBP (Ours)	**30.72**
	FIT: TRec (Ours)	25.99
	Ablation: Only FBP	30.49
	Ablation: Only Conv-Block	26.92
LoDoPaB [17] (downscaled)	Baseline: FBP	26.89
	FIT: TRec + FBP (Ours)	**30.98**
	FIT: TRec (Ours)	21.90
	Ablation: Only FBP	30.74
	Ablation: Only Conv-Block	30.70

Table 1. **Quantitative tomographic reconstruction results.** We report the average peak signal-to-noise ratio (PSNR) with respect to ground truth, for each of the three used datasets. For each dataset, we compare the results of our "FIT: TRec + FBP" and "FIT: TRec" setups to results obtained with the filtered backprojection (FBP) [11, 24] baseline, and the two ablation studies described in Section 4.5.

4.5. Training Setup for Computed Tomography

Like before, we consistently use $F = 256$ dimensional FDEs, and employ the *linear* encoder and decoder method by Katharopoulos *et al.* [12]. More specifically, we use 4 transformer layers, 8 self-attention heads per layer, a query and value dimensionality of 32, dropout of 0.1, attention dropout of 0.1, and a dimensionality of the feed-forward network of 1024. The residual conv-block has $d_{\text{conv}} = 8$ intermediate feature channels.

All networks are optimized using RAdam [18], with an initial learning rate of 0.0001 and weight decay of 0.01 for 300 (MNIST), 120 (Kanji), and 350 (LoDoPaB) epochs. The batch size is 32. We half the learning rate on plateauing validation loss.

Ablation Studies: We propose two ablation setups for all tomographic reconstruction experiments.

First, we ask what influence the encoded latent space information $\mathbf{Z}$, *i.e.* the output of the encoded sinogram, has on the quality of the overall reconstruction $\hat{\mathbf{x}}$. To that end, we perform ablation experiments for all 3 datasets, for which we do not feed $\mathbf{Z}$ to the decoder. Technically this is implemented by replacing the decoder by an encoder network (since only one input remains to be fed). We label these experiments "Only FBP".

The second ablation study asks, to what degree the conv-block contributes to the overall reconstruction performance,
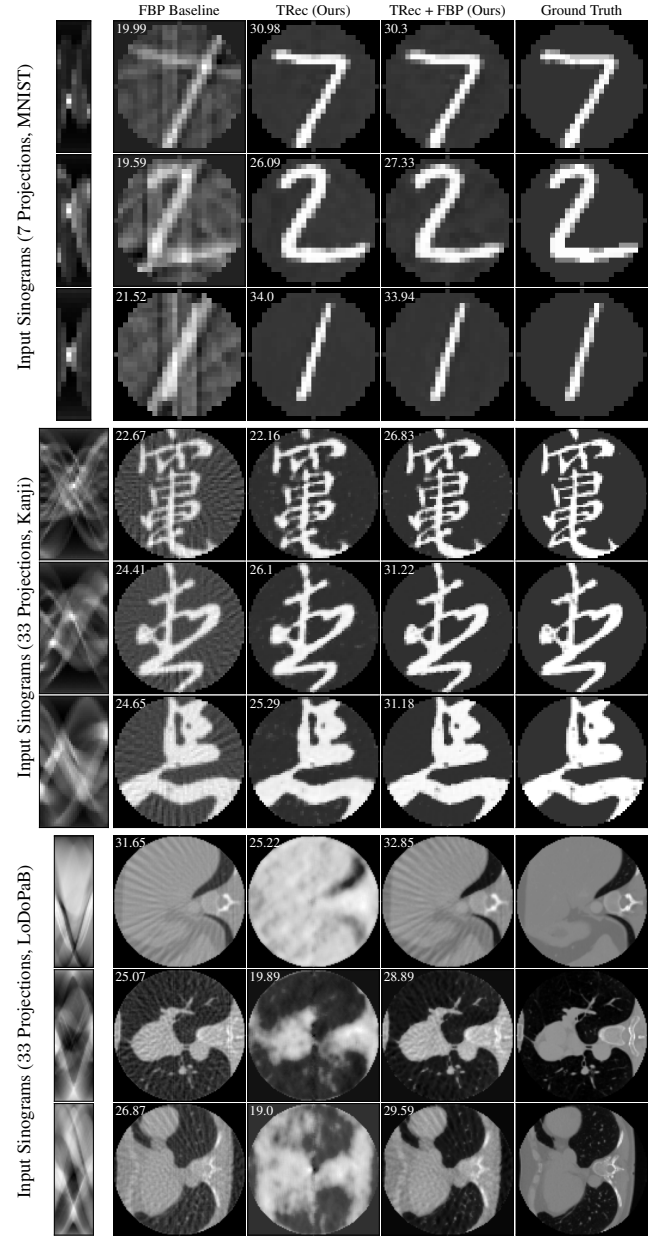

Figure 6. **Tomographic reconstruction results.** We show three qualitative results for each used datasets. From left to right, we show the input sinogram, reconstruction results obtained with filtered backprojection (FBP) [11, 24], our results obtained with the "FIT: TRec" setup, our results obtained with the "FIT: TRec + FBP" setup, and the corresponding ground truth images. In the top left corner of each reconstruction we show the peak signal-to-noise ratio (PSNR) with respect to the ground truth image.

i.e. we want to verify that the convolution block alone is not sufficient to solve the task at hand. Hence, we train the conv-block on pairs of images $(\text{FBP}(\boldsymbol{s}), \boldsymbol{x})$, *i.e.* the filtered backprojection of sinograms $\boldsymbol{s}$ and their corresponding ground truth images $\boldsymbol{x}$. We label these experiments

"Only Conv-Block".

For all ablation experiments all hyper-parameters not explicitly mentioned above are kept unchanged.

4.6. Results for Tomographic Reconstruction Tasks

Qualitative tomographic reconstruction results for all three datasets we used are shown in Figure 6. For each dataset, we show three input sinograms, the reconstruction baseline obtained via filtered backprojection (FBP), our results obtained via "FIT: TRec"and "FIT: TRec + FBP", and the corresponding ground truth images.

In Table 1, PSNR numbers for all three datasets using the FBP baseline, our "FIT: TRec + FBP" and "FIT: TRec" training setups, and both ablation studies are given.

5. Discussion

We proposed the idea of Fourier Domain Encodings (FDEs), a novel sequential image encoding, for which each prefix represents the whole image at reduced resolution, and demonstrated the utility of FDEs for solving two common image processing tasks with transformer networks, *i.e.* super-resolution and tomographic image reconstruction.

For the super-resolution task we showed that Fourier Image Transformer can be trained such that, when conditioned on an FDE corresponding to a low-resolution input image, can auto-regressively predict an extended FDE sequence that can be back-transformed into a higher resolution output. It is obvious, the information required to generate a higher resolution image must be stored in the trained network, and we have shown in Figure 5, how this learned prior completes very low to moderate resolution inputs in sensible ways. It is curious to see that eyes are the first high-resolution structures filled in by the trained FIT. We believe that this is a direct consequence of all training images being registered such that the eyes are consistently at the same location.

For the tomographic reconstruction task, we employ an encoder-decoder transformer that encodes a given FDE sequence corresponding to a given sinogram and can then be used to predict Fourier coefficients at arbitrary query locations. We used the decoder to predict all Fourier coefficients of a fully reconstructed image, which we could then visualize via inverse Fourier transformation (iFFT). We noticed that introducing a shallow residual convolution block after the iFFT reduces unwanted high frequency fluctuations in predicted results. While we see that this procedure leads to very convincing results on MNIST, for more complex datasets, results quickly deteriorate. Hence, we proposed to additionally feed Fourier coefficients obtained by filtered backprojection (FBP) into the decoder. This leads to much improved results that outperform the FBP baseline, showing that the FIT does contribute to solving the reconstruction task.

Our results show that transformers, currently the dominant approach for virtually all NLP tasks, can successfully be applied to complex and relevant tasks in computer vision. While this is encouraging, we see a plethora of possibilities for future improvements. For example, the transformers we used are rather small. We believe that an up-scaled version of our training setups with more attention heads and more layers would already lead to much improved results[4]. Still, we also believe that there is plenty of room for methodological improvements that do not require more computational resources, making this line of research also accessible to many other research labs around the globe.

Acknowledgments

References

[1] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE transactions on medical imaging*, 37(6):1322–1332, 2018. 3

[2] Irwan Bello. Lambdanetworks: Modeling long-range interactions without attention. In *International Conference on Learning Representations*, 2021. 2

[3] Ronald N Bracewell. Strip integration in radio astronomy. *Australian Journal of Physics*, 9(2):198–217, 1956. 3

[4] Hu Chen, Yi Zhang, Mannudeep K Kalra, Feng Lin, Yang Chen, Peixi Liao, Jiliu Zhou, and Ge Wang. Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE transactions on medical imaging*, 36(12):2524–2535, 2017. 3

[5] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 1, 2

[6] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature, 2018. 6, 7

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional

---

[4] https://tinyurl.com/needadviceforafriend
[5] https://www.youtube.com/c/YannicKilcher

transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1, 2

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2

[9] Andreas Hauptmann, Felix Lucka, Marta Betcke, Nam Huynh, Jonas Adler, Ben Cox, Paul Beard, Sebastien Ourselin, and Simon Arridge. Model-based learning for accelerated, limited-view 3-d photoacoustic tomography. *IEEE transactions on medical imaging*, 37(6):1382–1393, 2018. 3

[10] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017. 3

[11] Avinash C Kak, Malcolm Slaney, and Ge Wang. Principles of computerized tomographic imaging, 2002. 2, 3, 7

[12] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020. 1, 2, 6, 7

[13] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020. 2

[14] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. 5, 6, 7

[15] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 09 2019. 1

[16] James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*, 2021. 2

[17] Johannes Leuschner, Maximilian Schmidt, Daniel Otero Baguer, and Peter Maaß. The lodopab-ct dataset: A benchmark dataset for low-dose ct reconstruction methods. *arXiv preprint arXiv:1910.01113*, 2019. 6, 7

[18] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019. 6, 7

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 5

[20] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018. 2

[21] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training, 2018. 1, 2

[22] J. Radon. Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Akad. Wiss.*, 69:262–277, 1917. 2

[23] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[24] GR Ramesh, N Srinivasa, and K Rajgopal. An algorithm for computing the discrete radon transform with some applications. In *Fourth IEEE Region 10 International Conference TENCON*, pages 78–81. IEEE, 1989. 3, 7

[25] M Van Heel, W Keegstra, W Schutter, and EJF Van Bruggen. Arthropod hemocyanin structures studied by image analysis. *Life Chem. Rep. Suppl*, 1:69–73, 1982. 5

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 1, 2

[27] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020. 2

[28] Zelun Wang and Jyh-Charn Liu. Translating math formula images to latex sequences using deep neural networks with sequence-level training. *International Journal on Document Analysis and Recognition (IJDAR)*, pages 1–13, 2020. 2, 4