

Fast Fourier Intrinsic Network

Yanlin Qian¹, Miaojing Shi^{2*}, Joni-Kristian Kämäräinen¹, Jiri Matas³

¹Computing Sciences, Tampere University

²Department of Informatics, King's College London

³Center for Machine Perception, Czech Technical University in Prague

miaojing.shi@kcl.ac.uk

Abstract

We address the problem of decomposing an image into albedo and shading. We propose the Fast Fourier Intrinsic Network, FFI-Net in short, that operates in the spectral domain, splitting the input into several spectral bands. Weights in FFI-Net are optimized in the spectral domain, allowing faster convergence to a lower error. FFI-Net is lightweight and does not need auxiliary networks for training. The network is trained end-to-end with a novel spectral loss which measures the global distance between the network prediction and corresponding ground truth. FFI-Net achieves state-of-the-art performance on MPI-Sintel, MIT Intrinsic, and IIW datasets.

1. Introduction

Intrinsic image decomposition refers to the problem of separating an image \mathcal{I} into its *albedo* and *shading* [3], where *albedo* models the diffuse reflection of the scene materials and *shading* represents the lighting and scene geometry. We adopt the common assumption that the problem can be simplified to the per-pixel image formation $\mathcal{I} = \mathcal{A} \cdot \mathcal{S}$ where \mathcal{A} and \mathcal{S} are the albedo and shading, respectively. Factorization of \mathcal{I} into \mathcal{A} and \mathcal{S} is ill-posed. Early work tackled the factorization by applying constraints and priors which connect the properties of materials and lighting in the physical world with the gradient and color changes in images [21, 18]; however, even the sophisticated physical modelling can not fully cover the complex optical interactions among materials, light sources, and object/scene geometry. Intrinsic image decomposition is an open and challenging problem *per se*, but it can also benefit many other vision problems, *e.g.*, semantic segmentation [4], depth estimation [20], and color constancy [2].

Traditional approaches for intrinsic image decomposi-

tion are surveyed in [2], while recent state-of-the-art performance has been achieved with image-to-image translation networks as in [29, 13, 12, 23, 25, 27]. The mean squared error (MSE) loss is typically employed in the network training to enforce pixel-wise similarity between the predicted and ground truth albedo and shading [13, 23, 12]. The MSE loss does not consider correlations between image pixels. Extra loss terms such as loss for the learnable domain filter [13], perceptual loss [12] and generative adversarial loss [23], are often added to enforce global similarities between predictions and ground truth. Proposed models also exploit a multi-path structure in their networks to separate the prediction into different levels of details [12] or contexts [25, 13].

In this work, we propose a **Fast Fourier Intrinsic Network** (FFI-Net) for intrinsic image decomposition. The network design focuses on the multi-path architecture and loss function, as were intensively exploited in recent works [25, 13, 23, 12], but differs substantially from the prior arts since no published intrinsic decomposition method operates in the spectral domain. The intuition is based on the observation that published approaches [13, 12] often separate the predictions of structure and details into different scales, implicitly performing operations that are indeed best understood in the spectral domain. Image details, *e.g.* fine edges, are naturally reflected in the high-frequency part; on the other hand, image structures such as flattened color patches are usually reflected in the low-frequency part. Despite the frequency interpretation behind the previous works, their models were optimized in the spatial domain. Thus if the intrinsic image decomposition task is solved in the spectral domain, we might expect, and this is confirmed by our experiments, the optimization of the network to be more efficient and effective, as in the classification task [32]. Furthermore, since each pixel in the spectral domain is a complex-weighted sum over the image space, a pixel-wise loss on the spectral map thus can straightforwardly enforce a global similarity between the prediction and ground truth in the spatial domain.

*This work was partially done when Miaojing Shi was at INRIA Rennes and Yanlin Qian was an intern there.

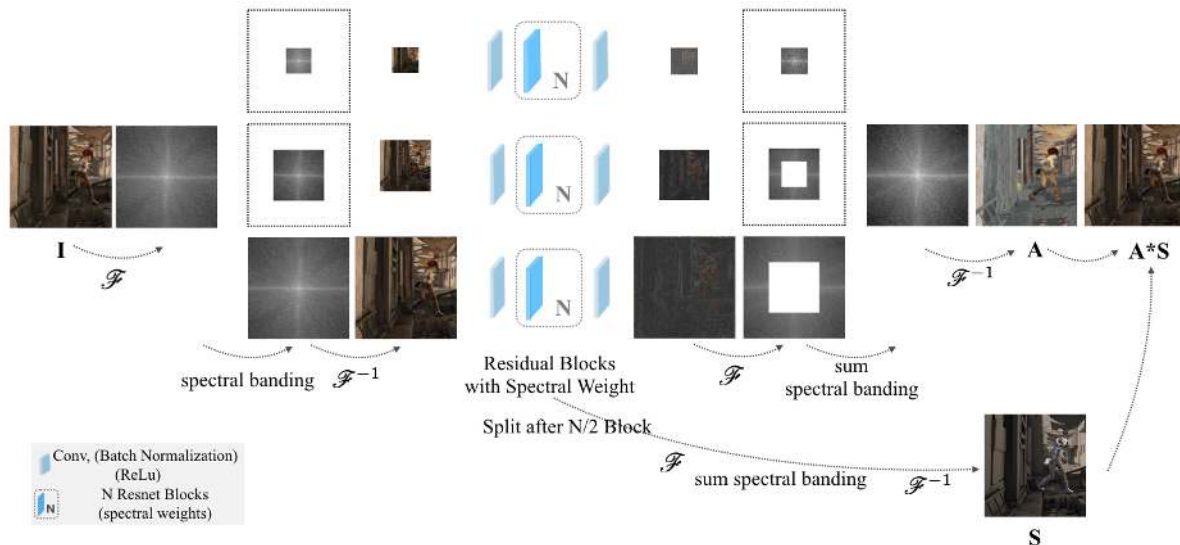


Figure 1: Overview of FFI-Net. Different from other deep intrinsic image decomposition architectures, FFI-Net offers several new modules: (a) \mathcal{F} and \mathcal{F}^{-1} layers that convert between spatial and spectral domains using the Fast Fourier Transform (FFT) and its inverse (Sec. 3.1); (b) *spectral residual blocks* where residual block weights are parametrized in the spectral domain (Sec. 3.2); (c) *spectral banding* that splits a spectral map into M (e.g. $M = 3$) overlapped (low-pass) spectral bands at the beginning of the network and associates M non-overlapped spectral bands of the image in both spatial and spectral domain as ground truth by the end of the network. (d) *spectral loss* which measures global error between the prediction and ground truth in the spectral domain (Sec. 3.5).

Contributions. We propose a novel network architecture for intrinsic image decomposition where the following spectral operations and structures replace their spatial domain counterparts used in the prior works:

- *Spectral banding* splits the input image into several frequency bands, therefore allowing multi-path training and inference (see Fig. 1).
- *Spectral loss* provides a global error measure on the similarity between the predicted A/S and the corresponding ground truth, effectively augmenting the pixel-wise loss.
- *Spectral residual block* encodes original residual block with spectral weights, thus obtaining faster convergence to a lower error.

To the best of our knowledge, FFI-Net is the first to address intrinsic decomposition in the Fourier domain. Our experiments with three intrinsic image datasets (MPI-Sintel, MIL Intrinsic and IIW) demonstrate state-of-the-art accuracy for FFI-Net.

2. Related Work

Our main objective is to develop a deep neural network architecture for intrinsic image decomposition and therefore related works will focus on the prior deep architectures [29, 20, 13, 23, 12, 5]. Traditional approaches have been surveyed in the recent work by Barron and Malik [2].

Nahiriha *et al.* [29] were the first to propose a deep ar-

chitecture to estimate albedo and shading using a pixel-wise MSE loss. Kim *et al.* [20] later added a conditional random field (CRF) after the network to jointly learn albedo, shading and depth. Lettry *et al.* [23] further proposed to add an adversarial module. To better model the flattening effects in natural reflectance images, Fan *et al.* [13] proposed a 1D recursive domain filter and separate loss layers customized differently for dense (e.g. MPI-Sintel [9]) and sparse (e.g. IIW [6]) datasets; Cheng *et al.* [12] proposed to use a cross bilateral filtering loss instead of the MSE loss to ensure the smoothness of albedo and shading output. Recently, Baslamisli *et al.* [5] adopted a physics-based image formation model inspired by traditional approaches.

The above works assume the Lambertian reflectance model and are fully-supervised with ground truth albedo and shading. There are also many works assuming a non-Lambertian model [34], adopting the weakly-supervised/unsupervised setting [19, 25, 28, 27], or including more synthetic data for training [24, 37]. For instance, Shi *et al.* [34] considered the non-Lambertian model in their algorithm design and train their network with rendered 3D models from ShapeNet [10]. Li *et al.* [25] learned intrinsic images from image sequences over time, where the object reflectance remains the same with varying illumination. Liu *et al.* [27] proposed a novel unsupervised intrinsic image decomposition framework which directly learns the latent feature of reflectance and shading from unsupervised and uncorrelated data. Zhou *et al.* [37] proposed a global-local

spherical harmonics lighting model for intrinsic decomposition, where they apply synthetic data for model pre-training and fine-tune it with real data in a self-supervised way.

The FFI-Net is inspired by several recent works while the most important being [12, 32]. FFI-Net differs from Cheng *et al.* [12] rather substantially in the following ways: (i) their hierarchical decomposition is done in the spatial domain with fixed-sized Gaussian kernels, while the decomposition in FFI-Net is made in the spectral domain; (ii) they use VGG perceptual loss to enforce global similarity between the intrinsic prediction and ground truth while FFI-Net uses the basic MSE in the spectral domain; (iii) their network optimization is done in the spatial domain while ours in the spectral domain. With respect to the comparison to [32], FFI-Net adopts the spectral encoding of CNN weights from it but novelly introduces several new elements *i.e.* spectral banding, spectral loss, spectral residual block, for intrinsic image decomposition.

3. Fast Fourier Intrinsic Network

Before presenting the building blocks of FFI-Net, we review the properties of the Discrete Fourier Transform.

3.1. Preliminaries

The Discrete Fourier Transform (DFT). For a discrete 2D image $I(h, w) = I \in \mathbb{R}^{H \times W}$ where $h = 0, \dots, H - 1$ and $w = 0, \dots, W - 1$, the Fourier domain representation is computed by

$$\mathcal{F}(I)_{nm} = \frac{1}{\sqrt{HW}} \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} I(h, w) e^{-2\pi i(\frac{nh}{H} + \frac{mw}{W})}$$

$$\forall n \in \{0, \dots, H - 1\}, \forall m \in \{0, \dots, W - 1\}. \quad (1)$$

The DFT is linear and unitary and I can be restored using the inverse DFT given by $\mathcal{F}^{-1}(I) = \mathcal{F}(I)^*$, which is the conjugate of the transform itself.

Properties of DFT for CNN.

Convolution theorem: Convolution in the spatial domain is equal to the element-wise product in the spectral domain:

$$\mathcal{F}(x * f) = \mathcal{F}(x) \odot \mathcal{F}(f), \quad (2)$$

where we denote by x and f two spatial signals (*e.g.* images), $*$ and \odot the operators of the convolution and the Hadamard product, respectively. DFT can be implemented using the Fast Fourier Transform (FFT), which reduces the complexity of DFT from $\mathcal{O}((HW)^2)$ to $\mathcal{O}(HW \log(HW))$.

DFT differentiation: For CNN backpropagation algorithms, the gradients must be propagated through Fourier transform layers. Suppose a DFT layer \mathcal{F} maps a real-valued input $x \in \mathbb{R}^{H \times W}$ to $z \in \mathbb{C}^{H \times W}$, and $\frac{\partial \mathcal{L}}{\partial z}$ is the gradient of loss

\mathcal{L} w.r.t z ; Since both \mathcal{F} and \mathcal{F}^{-1} are unitary, the back-propagation through \mathcal{F} and \mathcal{F}^{-1} can be written as

$$\frac{\partial \mathcal{L}}{\partial x} = \mathcal{F}^{-1} \left(\frac{\partial \mathcal{L}}{\partial z} \right), \quad \frac{\partial \mathcal{L}}{\partial z} = \mathcal{F} \left(\frac{\partial \mathcal{L}}{\partial x} \right). \quad (3)$$

3.2. The Network Architecture

Our network architecture is built upon the preliminary discussion about DFT and extended to the intrinsic image decomposition task with several new components (see Fig. 1). The input image I is first transformed into the spectral representation $\mathcal{F}(I)$. Spectral banding (Sec. 3.4) is applied afterwards to crop the spectral feature map of $\mathcal{F}(I)$ into M overlapped (low-pass) sub-representations. DC component is centered in the spectral feature map in Fig. 1. The M spectral bands are transformed back to the spatial domain via \mathcal{F}^{-1} as input to feed into N consecutive spectral residual blocks, respectively (Fig. 2 and Sec. 3.3). Spectral residual blocks parametrize the conv weights in the spectral domain for optimization yet their input and output are in the spatial domain. The outputs of the M sub-branches are transformed to the spectral domain, ground truth in two domains are associated both before and after the transformation. The M ground truth are generated via the spectral banding that is different from previous: it produces a series of non-overlapped bands similar to the Laplacian pyramid (Sec. 3.4). The M spectral maps predicted in the network are summed up to create a complete spectral map of the image and further converted back to the spatial domain; again, ground truth are associated both in the spectral and spatial domain on the complete image. The loss function between the ground truth and network predictions are introduced in the spatial and spectral domain, respectively (Sec. 3.5 and 3.6): the spatial loss employs the commonly used pixel-wise MSE in the original images and their gradient maps [13, 23] (Sec. 3.6); the spectral loss also utilizes pixel-wise MSE loss but on the spectral maps, which provides an additional global constraint on the intrinsic prediction (Sec. 3.5). Notice that, to infer both the albedo and shading simultaneously, after the $\frac{N}{2}$ -th block of each sub-branch we split it into two streams for albedo and shading, respectively.

Below we present the spectral residual block, spectral banding, and spectral loss.

3.3. Spectral Residual Block

Residual blocks [17] are widely applied for intrinsic image decomposition [13, 23]. Inspired by [32], we expect that spectral encoding of residual blocks can help solve the intrinsic decomposition task. As shown in Fig. 2, given a two-conv layer residual block, its input and output are in the spatial domain but the conv weights are parametrized in the spectral domain. These weights need to be converted back

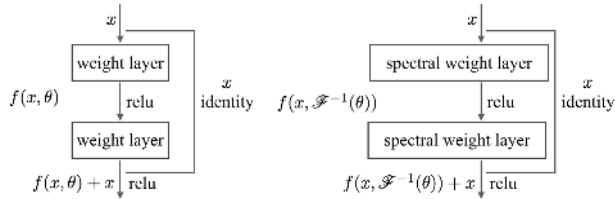


Figure 2: Illustration of the original (left) residual block and proposed spectral residual block (right). θ denotes the kernel weights of convolutional filters.

to the spatial domain when convolving with the input. This reparametrization does not alter the underlying model, it provides an appropriate domain for parameter optimization, as the frequency basis captures typical filter structure well. As suggested in [32], spectral encoding of the network results in more compact convolutional filters with more meaningful axis-aligned directions for optimization. In the experiment we demonstrate that residual blocks with spectral weights boost the performance.

3.4. Spectral Banding

Inspired by the multi-path structure in [12], which conducts scale space separation of the albedo/shading image, we instead perform a spectral domain separation, which provides more fine-grained control over frequency. The spectral banding is formulated as:

$$z_{nm} = \begin{cases} \mathcal{F}(x)_{nm}, & \text{if } n \in [N_l, N_u], m \in [M_l, M_u] \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

In a spectral map, values between N_l and N_u in x -axis, M_l and M_u in y -axis are kept. If N_l and M_l are always zeros, *spectral banding* works like a smoothing filter. We apply this special case of spectral banding (N_l and M_l are zeros while N_u and M_u are increased over bands) at the beginning of the network to split the input into overlapped bands (see Fig. 1 before the spectral residual blocks). If N_l and M_l are non-zeros, we can separate the spectral map into multiple bands where the upper bounds (N_u, M_u) of a current band are taken as the lower bounds (N_l, M_l) of its adjacent band with higher frequency. When we apply the inverse FFT to these spectral bands, they look like image Laplacian pyramid with different levels of details [8, 26] (see Fig. 1 after the spectral residual blocks). In the network, we generate M of these non-overlapped spectral bands from the ground truth to associate at the end of each subbranch. In this way, the subbranch output can be summed together to recover the final prediction (we call it *summing spectral bands* in Fig. 1).

In practice, we can provide a specific range of nm for $z_{nm} = 0$ in Eq. 4, where the spatial transform of a spectral band can be of adjustable size related to the predefined

range. This makes spectral banding flexible with any input and output size in a fully-convolutional neural network. Notice the level-of-detail exploited in [12] is obtained by applying the fix-sized Gaussian kernel in the spatial domain, while spectral banding retrieves the level-of-detail in the spectral domain explicitly and more effectively; Most importantly, our spectral banding enables the effective usage of another new component, spectral loss, as below.

3.5. Spectral Loss

Loss functions are defined in both the spectral and spatial domains, here we first detail the spectral loss: since each pixel in the spectral domain is a weighted sum of the whole image in the spatial domain, a pixel-wise loss defined on the spectral representation spontaneously enforces a global similarity between the spatial prediction and the ground truth. Inspired by the global loss applied in [23, 13, 12], we propose a spectral loss defined in the spectral domain:

$$\mathcal{L}_s(x, \hat{x}) = \|\mathcal{M}(\mathcal{F}(x)) - \mathcal{M}(\mathcal{F}(\hat{x}))\|_2^2 \quad (5)$$

where \hat{x} and x denote the predicted and ground truth albedo/shading in the spatial domain, respectively. \mathcal{M} is defined as an operator to take the magnitude of a complex value, $\mathcal{M}(\mathcal{F}(x)) = |\mathcal{F}(x)|$. We only use the magnitude part as it contains information of the image geometric structure in the spatial domain while discarding the phase part as “invisible” (see [1, 15]). This also empirically gives us benefits whilst minimizing the error over the phase part does not. Notice if we minimize the squared error for the real and imaginary components, since \mathcal{F} is a linear transform, it will be identical to minimizing the squared error in the spatial domain, as specified below.

In FFI-Net, spectral losses are directly applied pixel-wise on the M spectral outputs of the network as well as on their summation.

3.6. Network Training

Beside the *spectral loss*, we also adopt representative local MSE loss in the spatial domain to let the prediction to be both pixel-wise close and visually similar to the ground truth. Following [23, 13, 29], we choose the ℓ_2 loss defined on every pixel value and its gradient:

$$\mathcal{L}_p(x, \hat{x}) = \|x - \hat{x}\|_2^2, \quad \mathcal{L}_g(x, \hat{x}) = \|\nabla(x) - \nabla\hat{x}\|_2^2. \quad (6)$$

Each of the M subbranches in the network is trained with loss \mathcal{L}_{ss} from both spectral and spatial domains:

$$\mathcal{L}_{ss}(x, \hat{x}) = \lambda_1 \mathcal{L}_p + \lambda_2 \mathcal{L}_g + \lambda_3 \mathcal{L}_s. \quad (7)$$

We empirically choose $\lambda_1 = 0.2, \lambda_2 = 0.35, \lambda_3 = 10^{-6}$ for all our experiments. \mathcal{L}_s is of larger magnitude than \mathcal{L}_g and \mathcal{L}_p thus with a smaller weight λ_3 . \mathcal{L}_{ss} is applied

to all the M subbranches as well as their final summation in the network, so we have the overall loss $\mathcal{L}_{\mathcal{T}}$ for target albedo/shading:

$$\mathcal{L}_{\mathcal{T}} = \sum_{b=1}^M \mathcal{L}_{ss}^b(x_b, \hat{x}_b) + \mathcal{L}_{ss}(x, \hat{x}), \quad (8)$$

where we denote by x_b and \hat{x}_b the ground truth and prediction corresponding to the b -th spectral band of the original albedo/shading, respectively.

Referring to Fig. 1, $\mathcal{L}_{\mathcal{T}}$ is computed on both albedo and shading predictions ($\mathcal{L}_{\mathcal{A}}$ and $\mathcal{L}_{\mathcal{S}}$), so we have the final loss

$$\mathcal{L} = \mathcal{L}_{\mathcal{A}} + \mathcal{L}_{\mathcal{S}} + \mathcal{L}_{\mathcal{AS}} \quad (9)$$

$\mathcal{L}_{\mathcal{AS}} = \|\mathcal{A} \cdot \mathcal{S} - \hat{\mathcal{A}} \cdot \hat{\mathcal{S}}\|_2^2$ is added to address the commonly used constraint $\mathcal{I} = \mathcal{A} \cdot \mathcal{S}$ [12].

4. Experiments

Datasets. We conduct experiments on three popular datasets: *MPI-Sintel* [9], *MIT Intrinsic* [16] and *IIW* [6]. *MPI-Sintel* [9] consists of 18 animated video clips of multiple scenes, each containing 40 or 50 frames. We experiment with two different data splits: *image split* [29, 13, 12] and *scene split* [29, 12]. *MIT-Intrinsic* [16] contains 20 real objects in 11 different lighting configurations. The train/test split follows [2]. *IIW* [6] has 5,230 real images of mostly indoor scenes. Pairwise relative reflectance judgments are provided by humans as the ground truth albedo; the train/test split follows [13].

Training Details. For all datasets, we fix the number of spectral bands to $M = 5$ and use $N = 10$ spectral residual blocks for each band. Spectral residual blocks have the same configuration as the ordinary residual block [17] in terms of filter size, batch normalization, relu *etc.* Training is conducted using the Adam optimizer with network weights randomly initialized. (i) For *MPI Sintel*, we randomly crop 384×384 patch from each image. The upper and lower bounds (Eq.4) for each spectral band are $(384/4, 384/2)$, $(384/8, 384/4)$, $(384/16, 384/8)$, $(384/32, 384/16)$ and $(0, 384/32)$, where $N_l = M_l$ and $N_u = M_u$ for square crops. We set the batch size to 4 and the learning rate to $4e-4$. (ii) For *MIT Intrinsic* with smaller images, we randomly crop 192×192 patches and flip them. The upper and lower bounds for each band are $(192/4, 192/2)$, $(192/8, 192/4)$, $(192/16, 192/8)$, $(192/32, 192/16)$ and $(0, 192/32)$, respectively. The batch size is 8 and learning rate is $8e-3$. (iii) For *IIW*, we resize each image to 300×300 as in [13]; the batch size is set to 1 and the learning rate is $1e-2$. All networks are trained till convergences in our experiments.

Evaluation Protocol. We report results using the four common metrics: (i) *si-MSE*: scale-invariant mean squared

<i>Sintel scene split</i>	si-MSE $\times 10^2$			si-LMSE $\times 10^2$			DSSIM $\times 10^2$		
	A	S	avg	A	S	avg	A	S	avg
DI [29]	2.01	2.24	2.13	1.31	1.48	1.39	20.73	15.94	18.33
DARN [23]	1.77	1.84	1.81	0.98	0.95	0.97	14.21	14.05	14.13
Fan et al. [13]	1.81	1.75	1.78	1.22	1.18	1.20	16.74	13.82	15.28
LapPyrNet [12]	1.38	1.38	1.38	0.92	0.93	0.92	8.46	9.26	8.86
FFI-Net	1.19	1.16	1.18	0.73	0.73	0.73	8.93	8.60	8.77
M=5, R_{spec}	1.19	1.16	1.18	0.73	0.73	0.73	8.93	8.60	8.77
M=3, R_{spec}	1.24	1.16	1.20	0.75	0.75	0.75	9.59	8.85	9.22
M=1, R_{spec}	1.35	1.36	1.35	0.82	0.82	0.82	10.43	10.84	10.63
M=5, R_{spat}	1.50	1.50	1.50	0.91	0.92	0.91	10.25	9.78	10.01
M=3, R_{spat}	1.41	1.44	1.42	0.82	0.93	0.88	10.59	9.95	10.27
M=1, R_{spat}	1.52	1.67	1.60	0.91	1.14	1.02	11.04	11.69	11.37
w/o $\mathcal{L}_{\text{spec}}$ (M=5, R_{spec})	1.19	1.16	1.18	0.75	0.75	0.75	9.55	8.70	9.13
only $\mathcal{L}_{\text{spec}}$ (M=1, R_{spat})	1.29	1.28	1.29	0.77	0.84	0.81	10.11	9.13	9.62

Table 1: Results for *MPI-Sintel* (scene split). si-MSE, si-LMSE and DSSIM are reported for albedo (A), shading (S) and their average (avg). We also ablate the key elements of FFI-Net: number of spectral bands $M \in \{1, 3, 5\}$, spectral (R_{spec}) vs. spatial (R_{spat}) residual blocks, and the spectral loss $\mathcal{L}_{\text{spec}}$. FFI-Net is by default with $M=5$, R_{spec} . The best results are marked in bold.

error (MSE); (ii) *si-LMSE*: scale-invariant local MSE, which uses sliding windows in si-MSE; the window size w is usually 10% of the larger dimension of the image while the sliding step is $0.5w$; (iii) *DSSIM*: dissimilarity structural similarity index measure; it is made to correlate with the subjective image quality assessments [35]. For *MIT*, we also report the MSE and LMSE numbers which can be seen as variants of si-MSE and si-LMSE [16]. For *IIW* with sparse ground truth, we report the weighted human disagreement rate (WHDR) [6, 13].

We compare our method with a number of prior arts; these approaches are trained in various ways including fully-/weakly-supervised learning [13, 12, 24, 23, 28, 37], unsupervised learning [27, 31, 6, 7], as well as learning from auxiliary datasets [29, 34, 24, 37] (see Sec. 2). We do not elaborate every detail with every method but mainly focus our comparison with those representative fully-supervised methods, *e.g.* [13, 12, 24, 23].

4.1. Results on *MPI-Sintel*

Results for FFI-Net and various competing works are shown in Tab. 1 (scene-split) and Tab. 2 (image-split).

The scene-split setting is more challenging as it requires generalization over unseen scenes. Tab. 1 shows that FFI-Net performs the best on most metrics. In particular, it achieves the best average si-MSE, si-LMSE, and DSSIM of albedo and shading: 1.18, 0.73, and 8.77, respectively.

FFI-Net achieves the best performance also for the image-split setting reported in Tab. 2, but with smaller margin. We suggest a possible reason as the overfitting of deep networks on the shared contents between training and test images in image-split. This is illustrated by the recent re-

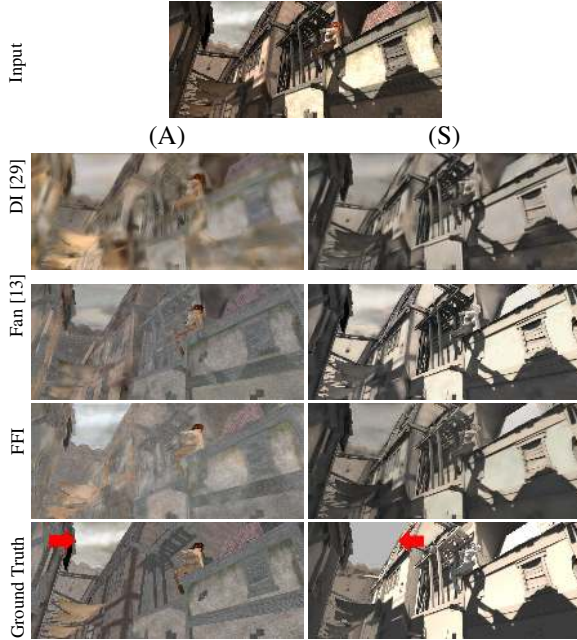


Figure 3: Examples on the MPI-Sintel (scene split). We emphasize the ability of FFI-Net of shadow removal (e.g. sky - red arrows) in the unseen scene compared to others.

Sintel <i>image split</i>	si-MSE $\times 10^2$			si-LMSE $\times 10^2$			DSSIM $\times 10^2$		
	A	S	avg	A	S	avg	A	S	avg
Retinex [31]	6.06	7.27	6.67	3.66	4.19	3.93	22.70	24.00	23.35
Lee et al. [22]	4.63	5.07	4.85	2.24	1.92	2.08	19.90	17.70	18.80
SIRFS [2]	4.20	4.36	4.28	2.98	2.64	2.81	21.00	20.60	20.80
Chen et al. [11]	3.07	2.77	2.92	1.85	1.90	1.88	19.60	16.50	18.05
DI [29]	1.00	0.92	0.96	0.83	0.85	0.84	20.14	15.05	17.60
DARN [23]	1.24	1.28	1.26	0.69	0.70	0.70	12.63	12.13	12.38
Kim et al. [20]	0.70	0.90	0.70	0.60	0.70	0.70	9.20	10.10	9.70
Fan et al. [13]	0.67	0.60	0.63	0.41	0.42	0.41	10.50	7.83	9.16
LapPyrNet [12]	0.66	0.60	0.63	0.44	0.42	0.43	6.56	6.37	6.47
Liu <i>et al.</i> [27]	1.59	1.48	1.54	0.87	0.81	0.84	17.97	14.74	16.35
FFI-Net	0.64	0.57	0.61	0.46	0.39	0.42	7.73	6.28	7.01

Table 2: Results for MPI-Sintel (image split).

sults in [20, 13, 12] from which the error numbers are indeed close to 0 (see Tab. 2, they are multiplied by 100).

LapPyrNet [12] performs in general the second best. It employs several loss functions such as joint bilateral filtering loss, VGG perceptual loss, *etc.* Notwithstanding, our results show that the proposed simple spectral loss (Tab. 1, only $\mathcal{L}_{\text{spec}}$) is as effective as these multiple losses.

Qualitative illustrations are presented in Fig. 3 and Fig. 4 where in particular the low-frequency part of the sky and high frequency details of the person’s face illustrate the improved prediction thanks to the spectral structure.

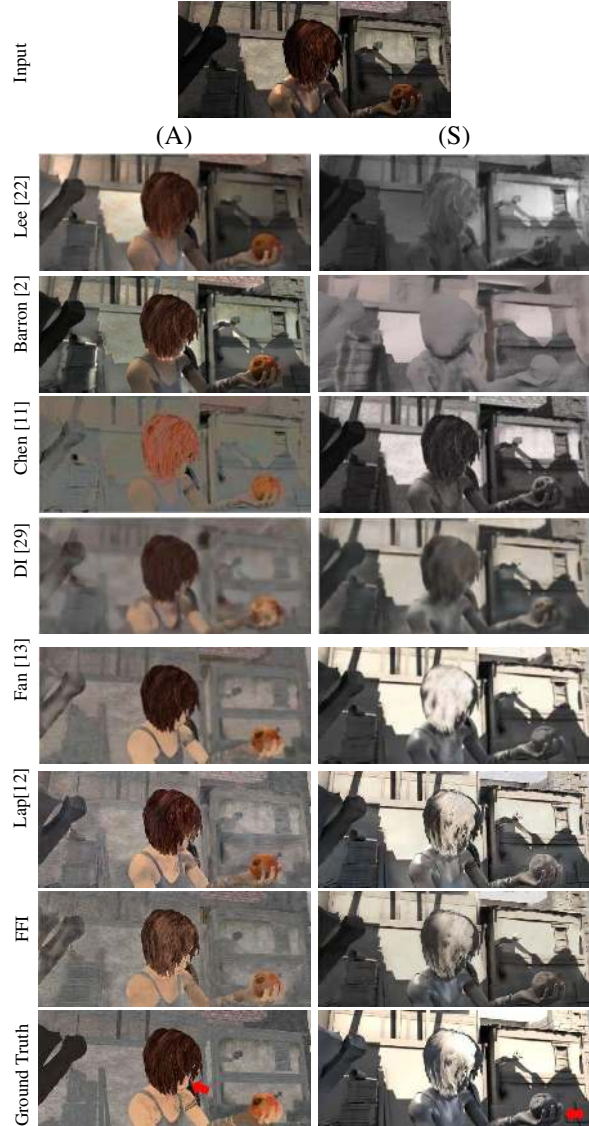


Figure 4: Examples of (A)lbedo and (S)hading predictions on MPI-Sintel (image split). FFI-Net performs well in difficult parts, e.g. person fingers, faces (red arrows).

4.2. Results on MIT Intrinsic

For MIT Intrinsic, [23, 13, 12] report si-MSE and si-LMSE and while [24, 5, 25] report MSE, LMSE and DSSIM. We include both in Tab. 3 and 4 for comprehensive comparison.

With si-MSE and si-LMSE (Tab. 3), FFI-Net outperforms the competing deep learning methods. The traditional method SIRFS [2] has slightly better si-MSE for albedo and total si-LMSE. It uses a prior that is particularly suitable for the MIT Intrinsic images; we notice its clear inferiority over others on MPI-Sintel in Tab. 2. MIT is a rather small dataset and is therefore often trained with additional datasets [34, 29] while FFI-Net only trains on MIT.

	Tr. data	si-MSE $\times 10^2$		si-LMSE $\times 10^2$	
		A	S	avg	Total
SIRFS [2]	M	0.64	0.98	0.81	1.25
Zhou <i>et al.</i> [38]	M	2.52	2.29	2.40	3.19
Shi <i>et al.</i> [34]	SN+M	2.16	1.35	1.75	2.71
DI [29]	ST+M	2.07	1.24	1.65	2.39
Fan <i>et al.</i> [13]	M	1.34	0.89	1.11	2.03
LapPyrNet [12]	I+M	0.89	0.73	0.81	1.41
FFI-Net	M	0.71	0.76	0.73	1.31

Table 3: Results for MIT Intrinsic. All methods use the training-test split file released in [2]. For si-MSE and si-LMSE, we use the evaluation code from [13, 2]. Tr. data lists the datasets used in training: MIT (M), ShapeNet (SN), Sintel (ST) and ImageNet (I). The best CNN-based results are marked in bold.

	Tr. data	MSE		LMSE		DSSIM	
		A	S	A	S	A	S
SIRFS [2]	M	1.47	0.83	4.16	1.68	12.38	9.85
DI [29]	ST+M	2.77	1.54	5.86	2.95	15.26	13.28
Shi <i>et al.</i> [34]	SN+M	2.78	1.26	5.03	2.40	14.65	12.00
CGI [24]	CGI	2.21	1.86	3.49	2.59	17.39	16.52
CGI [24]	CGI+M	1.67	1.27	3.19	2.21	12.87	13.76
Ma <i>et al.</i> [28]*	M	1.68	0.93	<i>0.74</i>	<i>0.52</i>	–	–
Liu <i>et al.</i> [27]	M	1.57	1.35	1.46	2.31	–	–
FFI-Net	M	1.11	0.93	2.91	3.19	10.14	11.39

Table 4: More results for MIT Intrinsic under different metrics, MSE, LMSE, and DSSIM ([28]* uses a different window size for LMSE). The best CNN-based results are marked in bold.

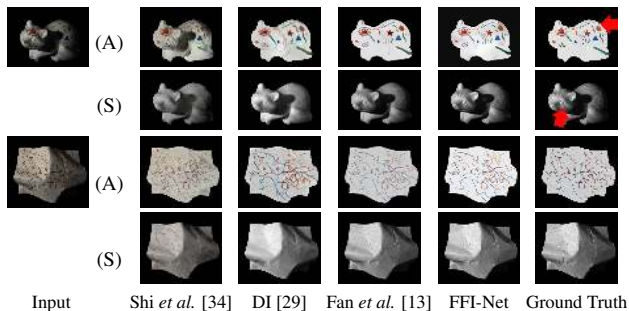


Figure 5: Sample (A)bedo and (S)hading on MIT Intrinsic. FFI-Net performs better for object details (top), as well as for the overall appearance (bottom).

We also provide results in MSE, LMSE, and DSSIM in Tab. 4. FFI-Net performs in general better than latest works *e.g.* [28, 24, 27]. Notice that it appears that the LMSE in [28] is computed using a different window size, thus its two numbers (italic) in LMSE are not comparable to others.

Methods	Training set	WHDR (mean)
DI [29]	ST	37.30
Shen <i>et al.</i> [33]	–	36.90
Retinex (color) [16]	–	26.89
Retinex (gray) [16]	–	26.84
Garces <i>et al.</i> [14]	–	25.46
Zhao <i>et al.</i> [36]	–	23.20
L_1 flattening [7]	–	20.94
Bell <i>et al.</i> [6]	–	20.64
Zhou <i>et al.</i> [38]	IIW	19.95
Nestmeyer <i>et al.</i> CNN[30]	IIW	19.49
Zoran <i>et al.</i> * [39]	IIW	17.85
Nestmeyer <i>et al.</i> [30]	IIW	17.69
Bi <i>et al.</i> [7]	–	17.67
CGI [24]	CGI+IIW	17.50
CGI [24]	CGI+IIW(A)+SAW	15.50
Fan <i>et al.</i> [13]	IIW	14.45
Zhou <i>et al.</i> [37]	SUNCG + IIW	15.20
Liu <i>et al.</i> [27]	IIW	18.69
FFI-Net w/ WHDR loss	IIW	15.81

Table 5: Results for IIW. Evaluation follows the train-test split in [29]. Note that due to the sparse ground truth available, FFI-Net does not use the spectral banding and loss, but instead the WHDR loss in [13].

4.3. Results on IIW

The IIW dataset differs substantially from MPI-Sintel and MIT Intrinsic in its ground truth. IIW does not have dense pixel-wise ground truth with physical grounds, but only sparse pair-wise annotations of points in each image. Therefore the losses relying on dense prediction, *e.g.*, GAN discriminator loss/VGG loss, in [12, 23, 28, 34] cannot apply. That also prevents using our spectral loss and sub-branch losses. For this experiment, our proposed loss functions were replaced with the WHDR loss used in [13]. Our network structure remains encoded with spectral weights.

The WHDR scores are shown in Tab. 5. FFI-Net, despite being disabled with spectral banding and loss, still achieves competitive WHDR score 15.81 among the state-of-the-arts [24, 13, 37]. The best performance by [13] can be partly explained by its tailored domain filter that flattens albedo prediction. While [24] and [37] use additional datasets (*e.g.* CGI, SAW, SUNCG) along with IIW for training augmentation.

4.4. Ablation Study

Spectral Bands. Tab. 1 demonstrates that adding more spectral bands (M) consistently improves the results. We use $M = 5$ as a trade-off between efficiency and accuracy. It is worth mentioning that using a single spectral band already outperforms the state-of-the-arts in si-MSE and si-



Figure 6: Qualitative comparisons of (A)bedo and (S)hading on IIW. Results of our FFI-Net are visually very similar to the best performing results [13].

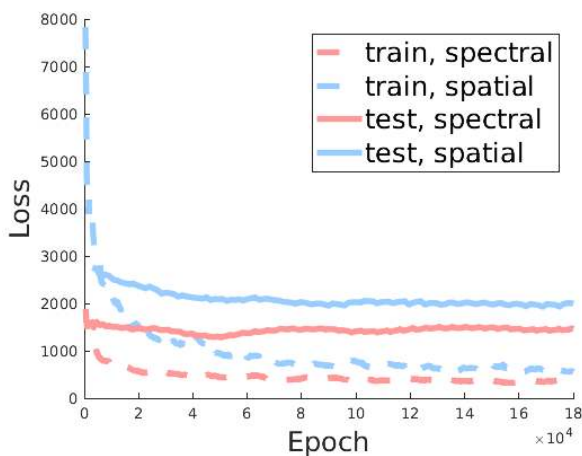


Figure 7: Network training convergence for the proposed spectral (red) and the conventional (blue) residual blocks.

LMSE. We believe this is due to the spectral encoding of residual blocks and the spectral loss.

Spatial vs Spectral Residual Blocks. Tab. 1 shows using spectral residual blocks performs significantly better than the spatial residual blocks (R_{spat} v.s. R_{spec}). Fig. 7 shows using spectral residual blocks significantly accelerates the convergence rate (3x - 5x) and improves the accuracy.

Spectral Loss. Tab. 1 reports the results of FFI-Net without spectral loss (w/o $\mathcal{L}_{\text{spec}}$): the performance is inferior to FFI-Net on the metrics si-LMSE and DSSIM but remains the same on si-MSE. Unsurprisingly, si-MSE defines the pixel-wise loss while si-LMSE and DSSIM consider the window-wise correlations. The results demonstrate the effectiveness of spectral loss on the global similarity between the intrinsic prediction and the corresponding ground truth. Moreover, we point out that FFI-Net (only $\mathcal{L}_{\text{spec}}$) already outperforms the state-of-the-arts [12, 23, 13] on si-MSE and si-LMSE.

Network parameters and inference time. Tab. 6 pro-

Method	Fan [13]	Darn [23]	Lap [12]	CGI [24]	FFI-Net
Parameters(Train)	2.80	17.97	138	7.7	0.75
Parameters(Test)	2.80	0.74	0.94	7.7	0.75

Table 6: Network parameter numbers displayed in million-scale. The training parameters for [23] and [12] includes that of the GAN discriminator and the VGG classifier, respectively.

files the network parameter number of FFI-Net. It generally needs less parameters compared to the others, which use extra guidance network [13], adversarial network [23], VG-GNet [12] and U-Net [24] for training. As for inference time, we reproduced [13, 12] with the same GPU (GTX 1080) as ours, it is ~ 100 ms for [13], ~ 70 ms for [12] and ~ 42 ms for FFI-Net.

5. Conclusion

We proposed a Fast Fourier Intrinsic Network architecture (FFI-Net) for intrinsic image decomposition. FFI-Net differs from the existing works in the sense that inference and network training is performed in the spectral domain and multi-scale learning is implemented by spectral banding. The spectral residual blocks introduced in this work converge faster and provide better accuracy than the conventional residual blocks. FFI-Net is lightweight and does not need auxiliary networks for training. Our experiments with three popular datasets verify that FFI-Net achieves state-of-the-art accuracy without auxiliary data training or specific parameter tuning. In the future work, we will explore the potential of spectral network components in other image-to-image translation tasks such as depth estimation and spatial illumination estimation.

Acknowledgement: This work was partially supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61828602.

References

- [1] Fourier transform. <https://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm>.
- [2] Jonathan T Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2015.
- [3] Harry Barrow, J Tenenbaum, A Hanson, and E Riseman. Recovering intrinsic scene characteristics. *Comput. Vis. Syst*, 2:3–26, 1978.
- [4] Anil S Baslamisli, Thomas T Groenestegge, Partha Das, Hoang-An Le, Sezer Karaoglu, and Theo Gevers. Joint learning of intrinsic images and semantic segmentation. In *ECCV*, 2018.
- [5] Anil S Baslamisli, Hoang-An Le, and Theo Gevers. Cnn based learning using reflection and retinex models for intrinsic image decomposition. In *CVPR*, 2018.
- [6] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)*, 33(4):159, 2014.
- [7] Sai Bi, Xiaoguang Han, and Yizhou Yu. An l1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. *ACM Transactions on Graphics (TOG)*, 34(4):78, 2015.
- [8] Peter Burt and Edward Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, 1983.
- [9] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.
- [10] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.
- [11] Qifeng Chen and Vladlen Koltun. A simple model for intrinsic image decomposition with depth cues. In *ICCV*, 2013.
- [12] Lechao Cheng, Chengyi Zhang, and Zicheng Liao. Intrinsic image transformation via scale space decomposition. In *CVPR*, 2018.
- [13] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David Wipf. Revisiting deep intrinsic image decompositions. In *CVPR*, 2018.
- [14] Elena Garces, Adolfo Munoz, Jorge Lopez-Moreno, and Diego Gutierrez. Intrinsic images by clustering. In *Computer graphics forum*, volume 31, pages 1415–1424. Wiley Online Library, 2012.
- [15] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.
- [16] Roger Grosse, Micah K Johnson, Edward H Adelson, and William T Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *ICCV*, 2009.
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
- [18] Berthold KP Horn. Determining lightness from an image. *Computer graphics and image processing*, 3(4):277–299, 1974.
- [19] Michael Janner, Jiajun Wu, Tejas D Kulkarni, Ilker Yildirim, and Josh Tenenbaum. Self-supervised intrinsic image decomposition. In *NIPS*, 2017.
- [20] Seungryong Kim, Kihong Park, Kwanghoon Sohn, and Stephen Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In *ECCV*. Springer, 2016.
- [21] Edwin H Land and John J McCann. Lightness and retinex theory. *Josa*, 61(1):1–11, 1971.
- [22] Kyong Joon Lee, Qi Zhao, Xin Tong, Minmin Gong, Shahram Izadi, Sang Uk Lee, Ping Tan, and Stephen Lin. Estimation of intrinsic image sequences from image+ depth video. In *ECCV*, 2012.
- [23] Louis Lettry, Kenneth Vanhoey, and Luc Van Gool. Darn: a deep adversarial residual network for intrinsic image decomposition. In *WACV*, 2018.
- [24] Zhengqi Li and Noah Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *ECCV*, 2018.
- [25] Zhengqi Li and Noah Snavely. Learning intrinsic image decomposition from watching the world. *CVPR*, 2018.
- [26] Wenqing Liu, Miaoqing Shi, Teddy Furon, and Li Li. Defending adversarial examples via dnn bottleneck reinforcement. In *ACM MM*, 2020.
- [27] Yunfei Liu, Yu Li, Shaodi You, and Feng Lu. Unsupervised learning for intrinsic image decomposition from a single image. In *CVPR*, 2020.
- [28] Wei-Chiu Ma, Hang Chu, Bolei Zhou, Raquel Urtasun, and Antonio Torralba. Single image intrinsic decomposition without a single intrinsic image. In *ECCV*, 2018.
- [29] Takuya Narihira, Michael Maire, and Stella X Yu. Direct intrinsics: Learning albedo-shading decomposition by convolutional regression. In *ICCV*, 2015.
- [30] Thomas Nestmeyer and Peter V Gehler. Reflectance adaptive filtering improves intrinsic image estimation. In *CVPR*, 2017.
- [31] E. H. Adelson R. Grosse, M. K. Johnson and W. T. Freeman. Ground-truth dataset and baseline evaluations for intrinsic image algorithms. In *ICCV*, 2009.
- [32] Oren Rippel, Jasper Snoek, and Ryan P Adams. Spectral representations for convolutional neural networks. In *NIPS*, 2015.
- [33] Li Shen and Chuohao Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. 2011.
- [34] Jian Shi, Yue Dong, Hao Su, and X Yu Stella. Learning non-lambertian object intrinsics across shapenet categories. In *CVPR*, 2017.
- [35] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- [36] Qi Zhao, Ping Tan, Qiang Dai, Li Shen, Enhua Wu, and Stephen Lin. A closed-form solution to retinex with non-

- local texture constraints. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1437–1444, 2012.
- [37] Hao Zhou, Xiang Yu, and David W Jacobs. Glosh: Global-local spherical harmonics for intrinsic image decomposition. In *ICCV*, 2019.
- [38] Tinghui Zhou, Philipp Krahenbuhl, and Alexei A Efros. Learning data-driven reflectance priors for intrinsic image decomposition. In *CVPR*, 2015.
- [39] Daniel Zoran, Phillip Isola, Dilip Krishnan, and William T Freeman. Learning ordinal relationships for mid-level vision. In *CVPR*, 2015.