SHS-GAN: Synthetic Enhancement of a Natural Hyperspectral Database

Jonathan Hauser, Gal Shtendel, Amit Zeligman, Amir Averbuch, and Menachem Nathan

***Abstract*—Deep Learning frameworks are gaining increased popularity in image processing tasks such as computational hyperspectral imaging. While these frameworks achieve state-of-the-art results in terms of reconstruction quality and run time, they often require massive databases of hyperspectral cubes for training the reconstruction algorithms. Unfortunately, such databases are usually hard to acquire due to complexity and cost considerations. To mitigate these challenges, we propose a method for generating a synthetic database of hyperspectral cubes in the visible range using a limited number of natural hyperspectral cubes, an unlimited number of RGB images, and a Generative Adversarial Network model. The suggested algorithm, dubbed SHS-GAN, is trained to get a query RGB image and to output a synthetic hyperspectral cube. While the spectral domain of the synthetic hyperspectral cube shares similar statistical properties as the natural hyperspectral cubes used in the training process, the SHS-GAN is trained to preserve the spatial characteristics of the query RGB image, whereas the R, G, B values provide an additional constraint along with the spectral domain. Our suggested framework was utilized for performing Snapshot Spectral Imaging (SSI) from a single monochromatic dispersed and diffused snapshot using the DD-Net reconstruction neural network. We demonstrate, by simulations and lab experiments, that enhancing the training database with synthetic data from the SHS-GAN improves the reconstruction quality of the hyperspectral cube. In addition, we share a new original database of more than 10,000 hyperspectral cubes of real objects of size 256x256x29 in the 420-700 nm visible range.**

***Index Terms*—Computational hyperspectral imaging, deep learning, image processing, neural networks.**

I. Introduction

RECENT years have witnessed tremendous growth in the usage of Deep Learning (DL) algorithms for a wide variety of computer vision and image processing tasks. Specifically, these algorithms have been proven to perform very well in various computer vision tasks such as image segmentation, classification, and detection [1], [2], as well as in image processing tasks such as depth reconstruction, denoising, and super-resolution [3]–[5]. As part of this progress, several DL methods were developed for the reconstruction of hyperspectral (HS) images from coded measurements obtained by various architectures [6]–[12]. It was demonstrated that DL methods can provide state-of-the-art results using architectures such as Coded Aperture Snapshot Spectral Imaging (CASSI) [7]–[9], which utilize Compressed Sensing (CS) algorithms for signal reconstruction. With part of this recent progress, recent work by our research team has demonstrated the reconstruction of HS cubes of size $256 \times 256 \times 29$ in the 420-700 nm spectral range from a single dispersed and diffused (DD) snapshot, acquired by a monochromatic camera with a pupil-domain phase diffuser, using the DD-Net algorithm [10].

Despite their promising performance in terms of reconstruction quality and runtime, a significant bottleneck of DL-based algorithms is the need for a massive database, required for the learning phase of the reconstruction Neural Network (NN). This challenge is especially prominent in HS imaging tasks, due to long and complex acquisition procedures and expensive equipment. While several databases of HS cubes are available online [13]–[16], their number is limited and may not embody the desired statistical properties in the spatial and spectral domains.

To overcome these challenges, it was suggested to enrich a database of real samples with synthetic samples [17], [18]. For example, it was demonstrated that synthetic HS data can contribute to different tasks such as Classification [19], Soil Detection [20], and Anomaly Detection [21].

A tempting framework for generating synthetic data is the Conditional Generative Adversarial Network (CGAN) [22]. Isola et al. [23], presented a CGAN based general framework for an adversarial image-to-image translation, which was proven to be efficient for natural image generation tasks [24], [25]. The CGAN framework, which is based on the GANs presented by Goodfellow et al. [26], offers a learning method for generating a synthetic database in a defined target domain. The synthetic output obeys a given statistical distribution, based on conditional input data from some source domain.

While some works present a framework for the generation of 1D spectra [27], [28], such information is not sufficient for HS reconstruction tasks, as spatial information is also required. In other methods, which are based on the CGAN framework, two separate Generators create the spectral and the spatial data in either parallel [29] or cascade [30] manner. However, since both networks are directed for improving HS classification, the generated HS data has a low spatial resolution and hence may

Manuscript received November 22, 2020; revised March 15, 2021; accepted April 24, 2021. Date of publication May 12, 2021; date of current version June 4, 2021. This work was supported in part by the Israeli Ministry of Science Technology and Space under Grants 3-10416, 3-13601, 3-16414, 3-14481, and by The Blavatnik Computer Science Foundation. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. James Theiler. *(Corresponding author: Jonathan Hauser.)*

Jonathan Hauser, Gal Shtendel, Amit Zeligman, and Menachem Nathan are with the Faculty of Engineering, Tel Aviv University, Tel Aviv 6997801, Israel (e-mail: hauserjo@post.tau.ac.il; galshten@mail.tau.ac.il; amitzeligman@mail.tau.ac.il; mnathan@tauex.tau.ac.il).

Amir Averbuch is with the Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 6997801, Israel (e-mail: amir1@tauex.tau.ac.il).

Digital Object Identifier 10.1109/TCI.2021.3079818

exhibit insufficient generation quality for training reconstruction algorithms in computational HS imaging architectures.

A possible way to improve spatial quality is to use readily available RGB images as priors. Lore et al. [31] presented a CGAN based framework, for learning RGB-to-multispectral mapping. Although their framework produces accurate HS cubes in both spectral and spatial domains, its structure is aimed at reconstruction. Thus, it is unlikely that this framework can generate synthetic HS cubes with desired statistical properties from a general RGB image query.

This paper describes a method of generating a large database of HS cubes using a Wasserstein GAN (WGAN) based NN architecture, denoted Synthetic Hyper-Spectral GAN (SHS-GAN). This network generates synthetic HS cubes in the visible spectral range that are both spectrally and spatially coherent with real HS cubes. To achieve this, a limited database of real HS cubes provides the statistical and spectral properties for the generated data. To achieve spatial coherency, the network is provided with real RGB images with similar spatial and color textures as the RGB representations of the real HS cubes. We used the synthetic database from the SHS-GAN to enhance a training database of the recently introduced DD-Net network [10], which enables us to reconstruct $256 \times 256 \times 29$ HS cubes in the 420-700nm visible range from a single dispersed and diffused (DD) monochromatic snapshot. We demonstrate, by simulations and lab experiments on real objects, that the quality of the reconstructed HS cubes can be improved by combining a small database of real data with a large database of synthetic data from the SHS-GAN.

The main contributions of this work are summarized as follows.

1) We present the SHS-GAN, which provides an end-to-end framework for enlarging HS databases based on RGB samples. To the best of our knowledge, there are no similar works of generating synthetic HS cubes in an adversarial manner, and hence we believe this work is novel.
2) We present a new, original HS database, which consists of 10666 HS cubes of real miniature objects of size $256 \times 256 \times 29$ in the 420-700 nm spectral range. The HS cubes database is shared in [32].
3) We demonstrate, by simulations and lab experiments on the real miniature objects, that synthetic database enhancement using the SHS-GAN can improve the quality of the DD-Net reconstruction algorithm.

Thanks to the modularity of the SHS-GAN framework, our demonstration provides a proof-of-concept for database enhancement for other computational HS imaging architectures. With minimal modification, it can be used to increase the proximity between a sub-domain captured by the training dataset and the full domain aimed to be learned by a data-driven algorithm. Consequently, we expect that the SHS-GAN may contribute to other HS-related tasks such as classification, segmentation, and object detection.

II. Generating Synthetic HS cubes Using WGAN

Following related works, we chose the WGAN presented by Arjovsky et. al in [33], with the Spectral Normalization

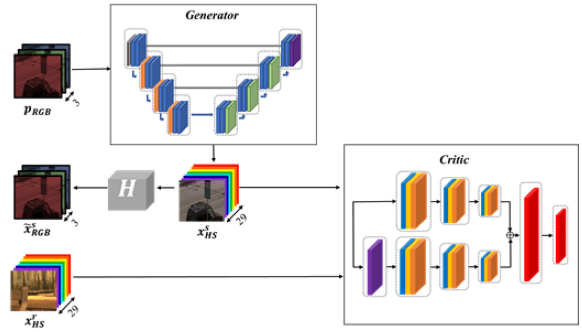

Fig 1. The SHS-GAN model.

GAN (SN-GAN) [34] that proved to be efficient for both the stability and the computational efficiency aspects of WGAN's training. WGAN architectures are composed of a Generator that produces synthetic samples, and a Critic that scores the realness or fakeness of a given sample.

Like [35], we decided to decouple the domain transfer process from the task-specific architecture to produce a more general framework, which allows one to artificially expand any general HS database for different purposes.

The scheme of our proposed SHS-GAN architecture is described in Fig. 1.

We define $\boldsymbol{p}_{RGB}$ to be an input image represented in RGB space. We would like to study a function $G(\boldsymbol{p}_{RGB}, \theta_G) \rightarrow \boldsymbol{x}^s_{HS}$, parametrized by θ_G, which learns a mapping between a source domain $\boldsymbol{X_{RGB}}$ and a target domain $\boldsymbol{X_{HS}}$, such that in the target domain the synthetic HS cube distribution $\boldsymbol{x}^s_{HS} \sim \mathbb{P}_{G(\theta_G)}$ and the real HS cube distribution $\boldsymbol{x}^r_{HS} \sim \mathbb{P}_r$ will be the same. Note that the spectral distribution we desire to approximate $\mathbb{P}_r$ has no spatial constraints, which can easily lead G to produce cubes that do not appear "natural". To overcome this, we use the prior $\boldsymbol{p}_{RGB}$ and encourage the output $\boldsymbol{x}^s_{HS}$ to be coherent with it. Therefore:

1. the spatial details of $\boldsymbol{p}_{RGB}$ are preserved in $\boldsymbol{x}^s_{HS}$, and
2. the color properties of $\boldsymbol{p}_{RGB}$ are preserved so objects in the cube keep their natural colors.

This model produces synthetic HS cubes that preserve the spatial features from the input images while adapting them to the HS domain based on a learned spectral distribution.

Although the real HS cubes can be converted to RGB images $\boldsymbol{x}^r_{RGB}$ deterministically, $\boldsymbol{x}^r_{RGB}$ and $\boldsymbol{p}_{RGB}$ are not drawn from the same distribution in the domain $\boldsymbol{X_{RGB}}$. Therefore, supplying the prior $\boldsymbol{x}^r_{RGB}$ to the network in a conditional manner will not produce the desired output. In our architecture, we use the prior $\boldsymbol{p}_{RGB}$ in a reconstruction loss manner, similarly to [36]: define $H : \boldsymbol{X_{HS}} \rightarrow \boldsymbol{X_{RGB}}$ to be a deterministic mapping from the HS to the RGB domain [37], and use $\tilde{\boldsymbol{x}}^s_{RGB} = H(G(\boldsymbol{p}_{RGB}))$ as the representation of the synthetic HS cube $G(\boldsymbol{p}_{RGB})$ in $\boldsymbol{X_{RGB}}$. We compare $\tilde{\boldsymbol{x}}^s_{RGB}$ and $\boldsymbol{p}_{RGB}$ and update the parameters θ_G according to this comparison.

A. Generator

The Generator architecture is shown in Fig. 2.

The Generator architecture is based on a modified version of the HS cube reconstruction network DD1-Net [10]. The

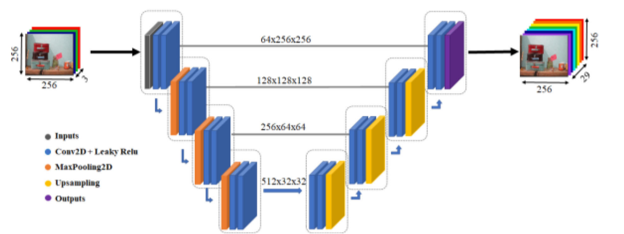

Fig 2. SHS-GAN Generator model. The output size of each block is written above the straight lines. The first number represents the number of features at this level. On each of the encoder's blocks the output is passed to the corresponding decoder stage via a skip connection.

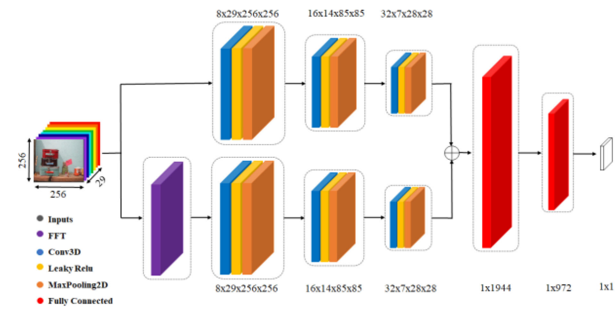

Fig 3. SHS-GAN Critic model. The output size of each block is written next to it. The first number represents the number of features at this level. The convolution kernels are: $(5,\ 3,\ 3)$, $(3,\ 3,\ 3)$, $(3,\ 3,\ 3)$, respectively.

DD1-Net architecture was chosen for the HS cubes Generator for two main reasons:

1. A U-net based architecture for WGAN Generators was proven to be efficient for image-to-image translations and style transfer tasks [23], [38].
2. The DD1-Net is designed to extract high-dimensional spectral data from low-dimensional spatial data.

Since our task does not focus on extracting deep features from the spatial dimension, but rather focuses on expanding the input's spectral dimension with quasi-natural properties, we use only a single bottleneck layer. In addition, the attention mechanism of the DD-Net was disabled.

1) Loss Function of the Generator: The objective of the Generator is to produce synthetic HS cubes that mimic samples from real HS cube distribution $\mathbb{P}_r$, while preserving spatial and color coherency of the real RGB images of the Generator's input. To achieve this, we introduce the following loss equation for the Generator: where L_G is the Generator loss, $\boldsymbol{p}_{RGB}$ is the input RGB image, $G(x^s_{RGB})$ is the synthetic HS cube from the Generator output, $\tilde{x}^s_{RGB}$ is the RGB representation of $G(x^s_{RGB})$, and $D(x_{HS})$ is the Critic output for a given HS cube x_{HS}. $y_{G(\boldsymbol{p}_{RGB})}$ is the corresponding label of $G(\boldsymbol{p}_{RGB})$, which denotes whether it is a real or synthetic HS cube, and if the Generator's training is 1 the Generator trains to "fool" the discriminator. λ_{RMSE} and λ_{SSIM} are the weight coefficients for the L_{RMSE} and L_{SSIM} loss metrics [10], [39], respectively. Hence, while the Generator is trained to produce HS cubes that will minimize the Critic loss, the other terms on the equation keep the RGB representation of the synthetic HS cube coherent with the input RGB image.

B. Critic

The Critic judges how likely the Generator's outputs are generated from the same distribution as the real natural HS cubes. The Critic's architecture scheme is shown in Fig. 3. The architecture is based on parallel processing of the input HS cube. One arm exhibits the original HS cube, and the other arm exhibits the cube after Fast Fourier Transform (FFT) along the spectral dimension. The second representation emphasizes characteristics of the spectra such as smoothness and general shape. In each domain, a series of 3-dimensional (3D) convolution operations followed by non-linear activations are performed, which results in the compression of the HS cube into a low-dimensional multi-channel feature space. Then, the output feature maps from each arm of the network are summed together and fully connected layers are used to gradually reduce the output vector dimension to a single scalar. The use of 3D convolutions enables the Critic to extract both spatial and spectral regions from the input HS cubes, rendering the learnable features strongly "spectrally oriented".

III. Hyperspectral Imaging From a Monochromatic Dispersed and Diffused Snapshot

The motivation of this work is to upgrade the reconstruction quality of the DD-Net algorithm [10]. In particular, the SHS-GAN is used to enrich databases of natural HS cubes used during the learning phase of the DD-Net NN.

The schematic layout of the optical system used in DD-Net is shown in Fig. 4(a). DD snapshots are formed using a regular monochromatic camera with a pupil domain binary encoded phase diffuser having either 1D or 2D symmetry. The snapshots include a single sharp replica of the imaged scene along with additional DD image replicas, as illustrated in Figs. 4(b) and 4(c).

The DD-Net network performs an end-to-end HS reconstruction from a single DD monochromatic snapshot. Specifically, it is designed to leverage the spectral information encoded in the DD replicas while preserving the spatial texture. DD1-Net is adapted for DD snapshots acquired using the 1D diffuser and DD2-Net is adapted for DD snapshots acquired using the 2D diffuser.

$$L_G = \underbrace{\lambda_{RMSE} \cdot L_{RMSE}\left(\boldsymbol{p}_{RGB},\ \tilde{x}^s_{RGB}\right) + \lambda_{\text{SSIM}} \cdot L_{SSIM}\left(\boldsymbol{p}_{RGB},\ \tilde{x}^s_{RGB}\right)}_{Reconstruction\ Loss} \times \underbrace{\lambda_{CRITIC}\left(D\left(G\left(\boldsymbol{p}_{RGB}\right)\right) \cdot y_{G(\boldsymbol{p}_{RGB})}\right)}_{Adverserial\ Loss}, \quad (1)$$

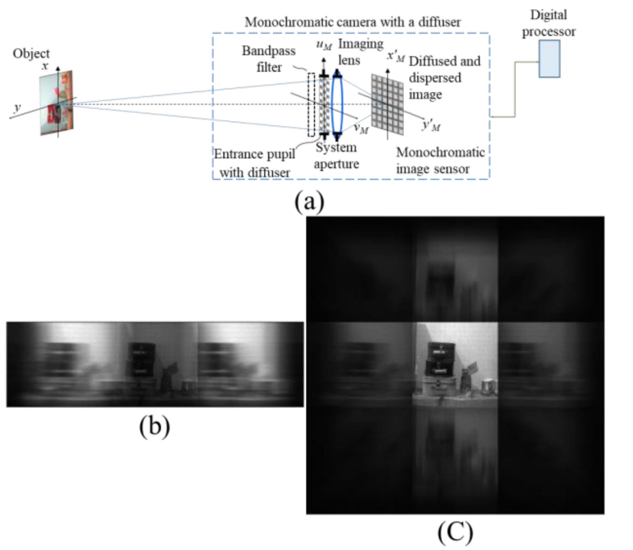

Fig. 4. (a) A Schematic layout of the optical system; (b), (c) DD snapshots obtained using a 1D or a 2D binary-encoded phase diffuser, respectively.

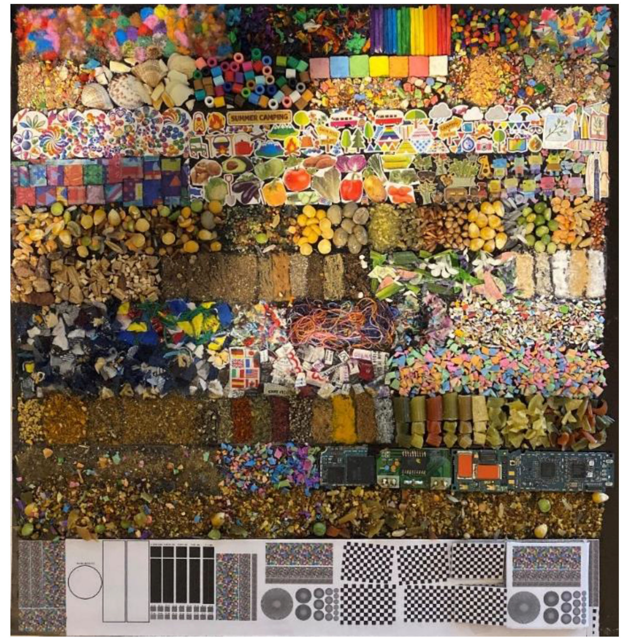

Fig. 5. The Objects Mosaic.

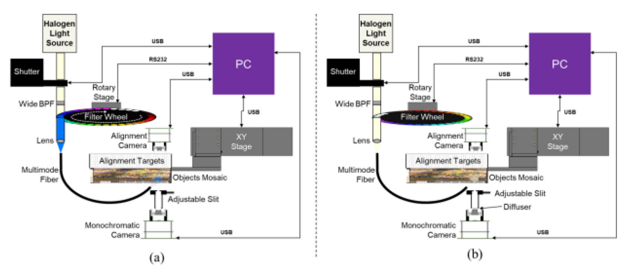

Fig. 6. Schemes of the experimental layout used for HS cubes and DD snapshots acquisition of multiple real objects. (a) HS cubes acquisition mode; (b) DD snapshots acquisition mode.

IV. Database Acquisition for Neural Network Training

A. Database Acquisition by Simulations

To test our system for reflective scenes and to assess our system performance, we used the ICVL HS database [13] which includes over 200 HS cubes of outdoor daylight scenes at a spatial resolution of 1392×1300 pixels. For each cube, 31 bands were sampled at the spectral interval of 400-700 nm with 10 nm increment. We cropped the cubes uniformly into $N_{y,i} \times N_{x,i} \times L = 256 \times 256 \times 31$ patches.

For training and validation data, we chose 7530 HS cubes such there is no overlap in the image environment between a training/validation set and the test set. We then simulated the sensor response to generate a DD snapshot measurement per HS cube. The simulation model was set according to each monochromatic camera configuration with either a 1D or a 2D binary phase diffuser.

B. Database Acquisition by Lab Experiments

Our previous research work [10] described an experimental method for obtaining a database of HS cubes and corresponding DD snapshots using an Apple iPad monitor as an object projector and a set of 29 narrow bandpass filters in the 420-700nm range. In this work, we constructed a lab setup for automatically acquiring thousands of HS cubes and corresponding DD snapshots of real reflective objects.

A custom "Objects Mosaic" was assembled from a large variety of multiple reflective objects with different colors and shapes. In particular, the mosaic included printed targets, colored chalk, beads, stickers, colored sponge, dried grains and legumes, dried pasta of different colors, spices, dietary fibers, synthetic and natural fabrics, soil samples, wood slivers in natural and artificial colors, artificial flowers, seashells, spices, electronic boards, paper shreds, and more. Fig. 5 shows an image of the Objects Mosaic, which has dimensions of 300 mm $\times$ 300 mm. Overall, 10666 HS cubes of size $256 \times 256 \times 29$ in the 420–700 nm spectral range were acquired using the monochromatic camera without the diffuser and a sequentially filtered Halogen light source, as illustrated in Fig. 6(a). The acquisition of corresponding DD snapshot images was done after placing the diffuser in front of the lens aperture, as illustrated in Fig. 6(b). To acquire polychromatic images, the filter wheel was rotated to an orientation in which no narrow bandpass filter is placed in the optical path of the illumination branch.

V. Generation of Synthetic HS Cubes and DD Snapshots for DD-Net Training

Our proposed method for HS database enrichment using the SHS-GAN was evaluated independently in two different experiments: a simulation experiment and a lab experiment. In both experiments, a real HS database was enriched using synthetic data from the SHS-GAN, and then the SSI system performance was evaluated with and without the additional synthetic data.

Fig. 7 summarizes the simulation and experimental frameworks for database enrichment and training of the DD-Net network using the SHS-GAN.

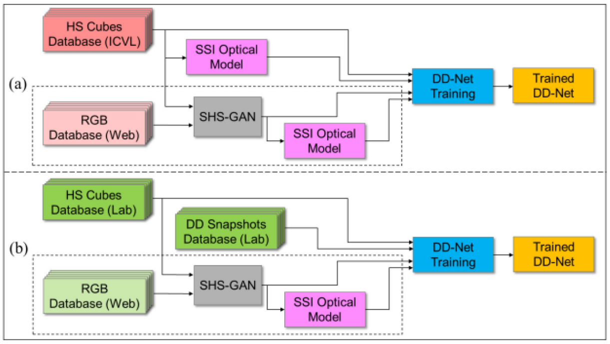

Fig. 7. Simplified block diagrams for (a) simulation and (b) lab experimental databases acquisition and enrichment for training the DD-Net NN.

Two computational units were used for the experiments:

I. A CPU with an Intel Core i-9 processor and 128 GB RAM installed with an Ubunto OS, and two Nvidia RTX 2080 TI GPUs.

II. A CPU with an Intel Core i-7 processor and 64 GB RAM installed with Windows10 OS, and a single Nvidia Quadro P2000 GPU.

The optimization of the Generator was performed using the ADAM algorithm [40] with a learning rate of 10^{-5}. The optimization of the Critic was performed using the RMSProp algorithm [41] with a learning rate $5 \cdot 10^{-5}$.

The simulation-based experiment was performed on the ICVL HS database [13], and the lab-based experiment was performed on the Objects Mosaic database. For each of these two natural HS databases, we created two different databases of natural RGB images found online that visually resemble the RGB representation of the natural HS cubes. The enriched ICVL database and the enriched Objects Mosaic database were used for training the DD-Net NN, for the evaluation of the SSI system by simulation and lab experiments, respectively.

A. Generating Synthetic Data From the ICVL Database

In the first experiment, synthetic HS cubes for our simulation experiment were generated using RGB databases of natural outdoor and urban scenes [42], [43] with similar spatial and color distribution as that of the ICVL Database [13]. The original RGB images were resized and then cropped to approximately 10000 patches of size 256 x 256 x 3. The patches were used for the training of the SHS-GAN along with only 1000 HS cubes of size $256 \times 256 \times 31$ in the 400-700nm spectral range, which were randomly sampled from the training database. After the SHS-GAN was trained, we generated 10000 synthetic HS cubes, which were then used to calculate the corresponding DD snapshot.

Before being fed into the Generator network, the dynamic range of each RGB image was transformed to the range $[-1,\ 1]$ to achieve more stable gradients and outputs and to cancel the undesired relative illumination differences between images from different sources or scenes. The HS cubes used in the Critic training were normalized globally over the whole HS database to be in the range of [0, 1]. Each spatial pixel of a cube was filtered independently along its spectral dimension with a Savitzky-Golay filter [44] with a polynomial order of 2 and a window width of 7. A representative ensemble of generated synthetic HS cubes is presented in Fig. 8.

The synthetic HS cubes were then used to calculate DD monochromatic snapshots for system configurations with either a 1D or a 2D binary-encoded phase diffusers.

B. Generating Synthetic Data From the Objects Mosaic Database

In the second experiment, an RGB database was assembled from different web sources that fit the large objects variety of the Objects Mosaic. The original images were zoomed and then cropped to patches of size $256 \times 256 \times 3$, so they spatially resemble the real HS cubes, albeit with a much larger variety of patterns, colors, orientations, and shapes. Accordingly, the SHS-GAN was trained with 20000 RGB images and 3000 lab-acquired HS cubes. The lab-acquired HS cubes were chosen to have a relatively large amount of spatial details, a high signal-to-noise ratio (SNR), and a minimal number of shading areas caused by the 3D topography of the real objects and the illumination arrangement. After the training of the GAN, we generated 20000 samples of synthetic HS cubes, and for each synthetic cube, we simulated a corresponding DD snapshot. Unlike the simulations, our lab experiments used only the 1D diffuser. Hence, the synthetic HS cubes were used for calculating DD snapshots considering only the camera configuration with the 1D diffuser. Before the training phase of this network, the same normalization and filtering procedures were performed as in the simulation-based experiment with the ICVL database.

During the experimental data acquisition, an adjustable slit was used as a field-of-view (FOV) blocker to avoid overlapping between the Image Replicas in the DD snapshot. To keep consistency during the entire lab experiment, we kept the FOV blocker also during the acquisition of the HS cubes. Due to its presence, our lab-acquired HS cubes were slightly vignetted near the left and right margins, as can be seen in the RGB representations of the synthetic HS cubes in Fig. 8.

C. Evaluation of the Synthetic HS Data

As seen in Fig. 8, the generated HS cubes preserve the spatial properties and colors of the input RGB images, while specific database-oriented features are generated by the SHS-GAN based on the natural HS database properties.

For the ICVL database, we note that the RGB images appear with a bright yellowish shade. This results from the relatively low-intensity values of the spectrum around the shorter wavelengths in the original database. As expected, the general characteristic trend of the spectra is preserved for each spatial pixel.

In the synthetic HS images produced by the SHS-GAN trained on our lab experimental database, the generator artificially introduced the vignetting effect mentioned in section B. This property is beneficial, as it allows to calculate synthetic DD snapshots which are more similar to the real DD snapshots.

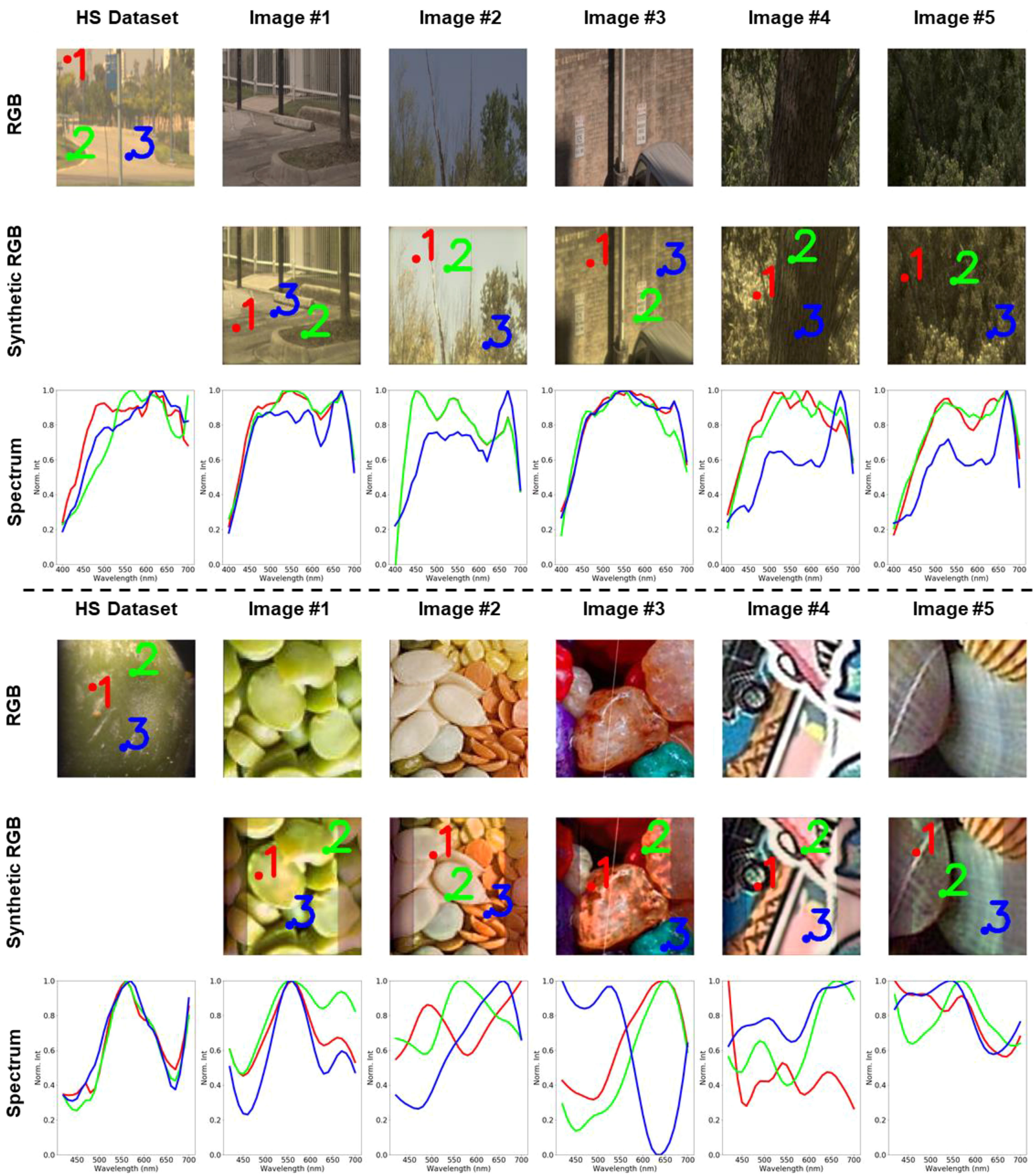

Fig 8. RGB representations and spectra of the synthetic HS cubes at selected spatial locations. UPPER: HS cubes synthesis based on the ICVL database; The similarity between Points 1 and 2 in Image #2 demonstrates that the SHS-GAN can generate coherent synthetic spectra in spatially homogeneous areas, such as clear sky. LOWER: HS cubes synthesis based on the Objects Mosaic database acquired in our laboratory. The leftmost image on each part is an RGB representation of a real, natural HS cube from the corresponding HS database. The color of each spectrum curve corresponds to the sampled point in the RGB image above it.

Further quantitative evaluation of the synthetic HS cubes was conducted on the lab experimental HS database as follows: 5 representative samples from the real database were chosen and 50 spectra were randomly sampled from each HS cube. For each of the real HS cubes, we selected 10 synthetic HS cubes that share some characteristics with the real example (colors, texture, etc.), and spectra were sampled from the same 50 points for each synthetic cube. We then calculated the mean spectral angle mapping (SAM) between each of the 50 real spectra and each of the 50 synthetic spectra and averaged the result over the 10 synthetic cubes. To evaluate the benefit of the Fast Fourier Transform (FFT) arm in the discriminator, we performed the

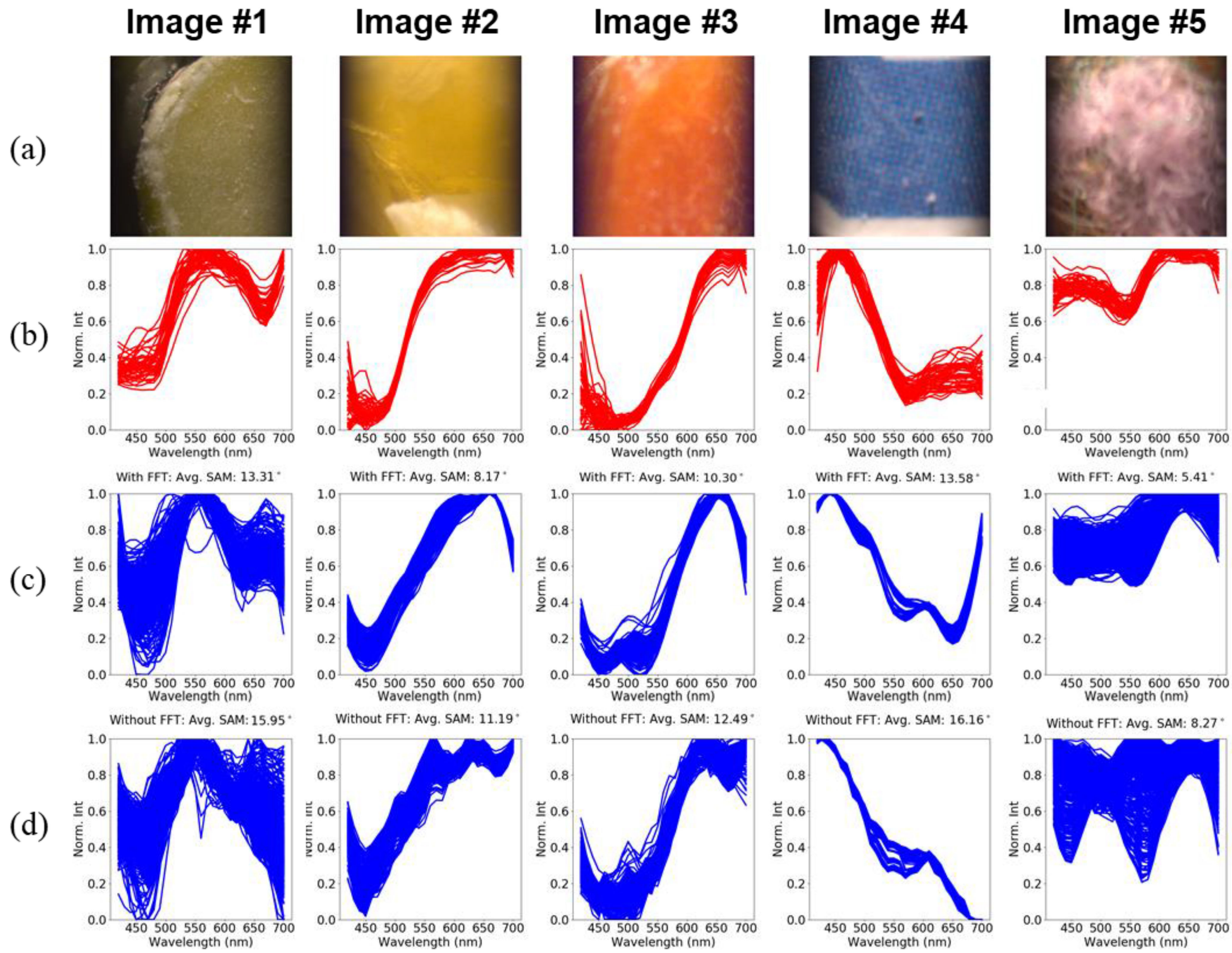

Fig 9. Real and synthetic spectra of 50 points with and without the FFT arm of the SHS-GAN discriminator. The titles display the average SAM between each sampled synthetic spectrum and each sampled real spectrum. (a) RGB representations of real, natural HS cubes from the Objects Mosaic database. (b) Spectra of 50 randomly sampled points in the real HS cube. (c) Spectra of 50 randomly sampled points in the synthetic HS cube generated by SHS-GAN. (d) Spectra of 50 randomly sampled points in the synthetic HS cube generated by SHS-GAN without the FFT arm.

same process twice: For the synthetic cubes produced by the SHS-GAN with the FFT arm in the discriminator, and without it. The correlation between the real and synthetic spectra of different materials is shown in Fig. 9, which demonstrates that the synthetic spectra is indeed regularized by the prior RGB image jointly with the real natural spectra. The contribution of the FFT arm to synthesis stabilization is also observed. As can be seen, the FFT arm forces the Generator to produce smoother and less noisy spectral curves which are more consistent with the real spectra, as reflected by the lower average SAM score they achieve. However, an interesting boundary effect can be observed for the longer wavelengths, where the network that was trained with the FFT arm shows a tendency to clip the values to a normalized intensity between 0.6–0.8. While the synthetic spectra share similar outlines as the natural spectra, it is not identical to it. Hence, enhancing the natural HS data with the synthetic HS data expands the variety of the training data, and strengthens the diversity and robustness of the DD-Net reconstruction algorithm.

VI. Reconstruction of HS Cubes From Monochromatic DD Snapshots

A. Evaluation on Simulated Data

Continuing the analysis reported in [10], we examined whether enhancing the training database of the DD-Nets with additional synthetic HS cubes and corresponding calculated DD snapshots indeed improves the reconstruction quality. The DD1-Net (for a system configuration with a 1D diffuser) and the DD2-Net (for a system configuration with a 2D diffuser) were trained with and without the additional 10000 synthetic HS cubes generated by the SHS-GAN. Specifically, for each system integrated with either a 1D or a 2D diffuser, we tested 4 training configurations: $(i.a)$ A test with all the available cubes from the ICVL database (7530 cubes), including augmentations as described in [10]; $(i.b)$ A test with all the available cubes from the ICVL database, including augmentations, and with the additional 10000 synthetic HS cubes from the SHS-GAN; $(ii.a)$ A test with 1000 HS cubes from the ICVL database,

TABLE I
SIMULATION RECONSTRUCTION RESULTS SUMMARY FOR 18 TEST CUBES, FOR SYSTEMS WITH A 1D AND A 2D DIFFUSER, FOR CONFIGURATIONS (*I*) AND (*II*).

Number of Natural HS cubes	System Configuration	Training Algorithm	PSNR [dB]			SSIM			SAM [deg]		
			Best	Worst	Mean	Best	Worst	Mean	Best	Worst	Mean
(i) 7530 Cubes	Mono Camera with 1D Diffuser	DD1-Net	**38.198**	19.298	29.919	0.985	0.884	0.953	1.381	6.403	3.213
		DD1-Net + SHS-GAN	38.128	**19.709**	**30.729**	**0.987**	**0.894**	**0.958**	**1.207**	**5.673**	**2.919**
	Mono Camera with 2D Diffuser	DD2-Net	41.500	**26.386**	**34.628**	0.992	**0.952**	**0.979**	1.025	4.862	2.272
		DD2-Net + SHS-GAN	**41.915**	22.845	34.582	**0.993**	0.948	0.979	**0.993**	**4.552**	**2.184**
(ii) 1000 Cubes	Mono Camera with 1D Diffuser	DD1-Net	30.444	18.966	25.519	0.940	0.803	0.880	2.286	9.249	4.490
		DD1-Net + SHS-GAN	**35.332**	**19.268**	**28.832**	**0.980**	**0.876**	**0.950**	**1.705**	**7.112**	**3.767**
	Mono Camera with 2D Diffuser	DD2-Net	31.964	19.675	26.204	0.978	0.902	0.950	2.854	9.514	5.569
		DD2-Net + SHS-GAN	**33.665**	**21.464**	**28.017**	**0.983**	**0.912**	**0.959**	**2.637**	**8.545**	**4.689**

including augmentations; $(ii.b)$ A test with 1000 HS cubes from the ICVL database, including augmentations, with additional 10000 synthetic HS cubes. The 1000 HS cubes in configurations $(ii.a)$, $(ii.b)$ are the same HS cubes used for the SHS-GAN training, as explained in Section V.

Table I summarizes the simulation results for each training configuration. The table indicates that the quality of the results is directly affected by the total number of natural HS cubes used in the learning phase of the DD-Net. While the total number of natural HS cubes plays the most crucial role in the overall reconstruction quality, the database enhancement with the synthetic HS cubes from the SHS-GAN contributes to further improvement for a given number of HS cubes used in the training. In particular, the contribution of the additional synthetic data plays a more significant role when the size of the original real database was limited to 1000 HS cubes. Hence, we believe that the SHS-GAN could be most useful for database enhancement when the number of natural HS cubes is limited. Note that configuration (i) with the DD2-Net algorithm already achieves state-of-the-art reconstruction performance, and thus is expected to be the hardest to improve. However, the integration of synthetic data from the SHS-GAN does improve the Best, Worst, and Average SAM values, which is considered to be a more significant metric for evaluation of HS reconstruction quality, and outperforms the recent simulation results achieved by our group [10]. We also note that the DD2-Net is generally expected to achieve better results than the DD1-Net, thanks to the processing of two additional dispersed and diffused image replicas along the vertical dimension of the monochromatic sensor [10]. However, we see an interesting result in which the DD1-Net outperforms the DD2-Net for training configuration (ii) (i.e., with 1000 natural HS cubes). This can be explained by the more compact architecture of the DD1-Net, which can be trained effectively from smaller data sets. Nevertheless, the contribution of the SHS-GAN is prominent for each system configuration.

Fig. 10 demonstrates the contribution of the SHS-GAN to the reconstruction quality for 3 exemplary scenes simulated for an optical system with a 2D diffuser, according to training configuration (i, b). Fig. 11 presents the root mean square error (RMSE) and SAM error maps of the reconstructed HS cubes.

B. Evaluation on Experimental Data

The test database for our lab experiments included 31 hand-chosen HS cubes that provided a representative sample of the entire Objects Mosaic database. The chosen HS cubes were isolated from the training database and neither the SHS-GAN nor the DD1-Net network were exposed to them. The lab experimental results are summarized in Table II.

Table II indicates that the reconstructions obtained with the SHS-GAN + DD1-Net framework achieve an average peak-signal-to-noise (PSNR) value of 25.466[dB], an average SSIM value of 0.958 and an average SAM value of 9.764, which outperforms the results without the SHS-GAN enrichment. It is worth mentioning that HS reconstruction of real experimental data is a more challenging task than reconstruction of simulated data, as it is subjected to various artifacts such as vignetting, optical system noise, registration errors between the HS cubes database and the DD cubes database, and optical model mismatches. These artifacts degrade the reconstruction quality, especially in regions with low SNR. Accordingly, spatial regions with higher SNR achieve better results with respect to the mean values. Fig. 12 presents the reconstruction of three test scenes from the Objects Mosaic. As seen in the spectral curves, database enhancement using synthetic cubes from the SHS-GAN significantly improves the reconstruction quality. Accordingly, the RGB reconstruction from the DD1-Net + SHS-GAN algorithm has better visual quality than the corresponding RGB reconstruction obtained by the DD1-Net algorithm. We wish to emphasize that all the scenes are of real, reflective objects, obtained solely with a monochromatic camera with a phase-only 1D binary diffuser. Scene 1 is of a sticker with a flat 2D topography, whereas Scenes 2 and 3 are of a green shred of paper and yellow peas, respectively, with a variable 3D topography. Fig. 13 presents the RMSE and SAM error maps of the reconstructed HS cubes.

A further analysis was conducted to evaluate the dependency of the reconstruction performance on the number of natural HS samples. We started with 600 real HS cubes randomly sampled from the training database of 3000 HS cubes and ran the whole training process for the SHS-GAN and the DD1-Net. The exact same process was then repeated four more times, each time with additional 600 random HS cubes added to the accumulating database. We used the same test database of 31 HS cubes in all the configurations. The analysis results are shown in Fig. 14.

Our analysis indicates that the DD1-Net + SHS-GAN framework is advantageous over the DD1-Net framework in terms of reconstruction quality in almost every figure of merit and any number of natural HS cubes used for training. A single exception was found in the PSNR figure of merit when the number of natural HS cubes was 600 (an average drop of -0.283 dB), however, the more meaningful SAM figure of merit was significantly improved (an average drop of -1.973 degrees), as well

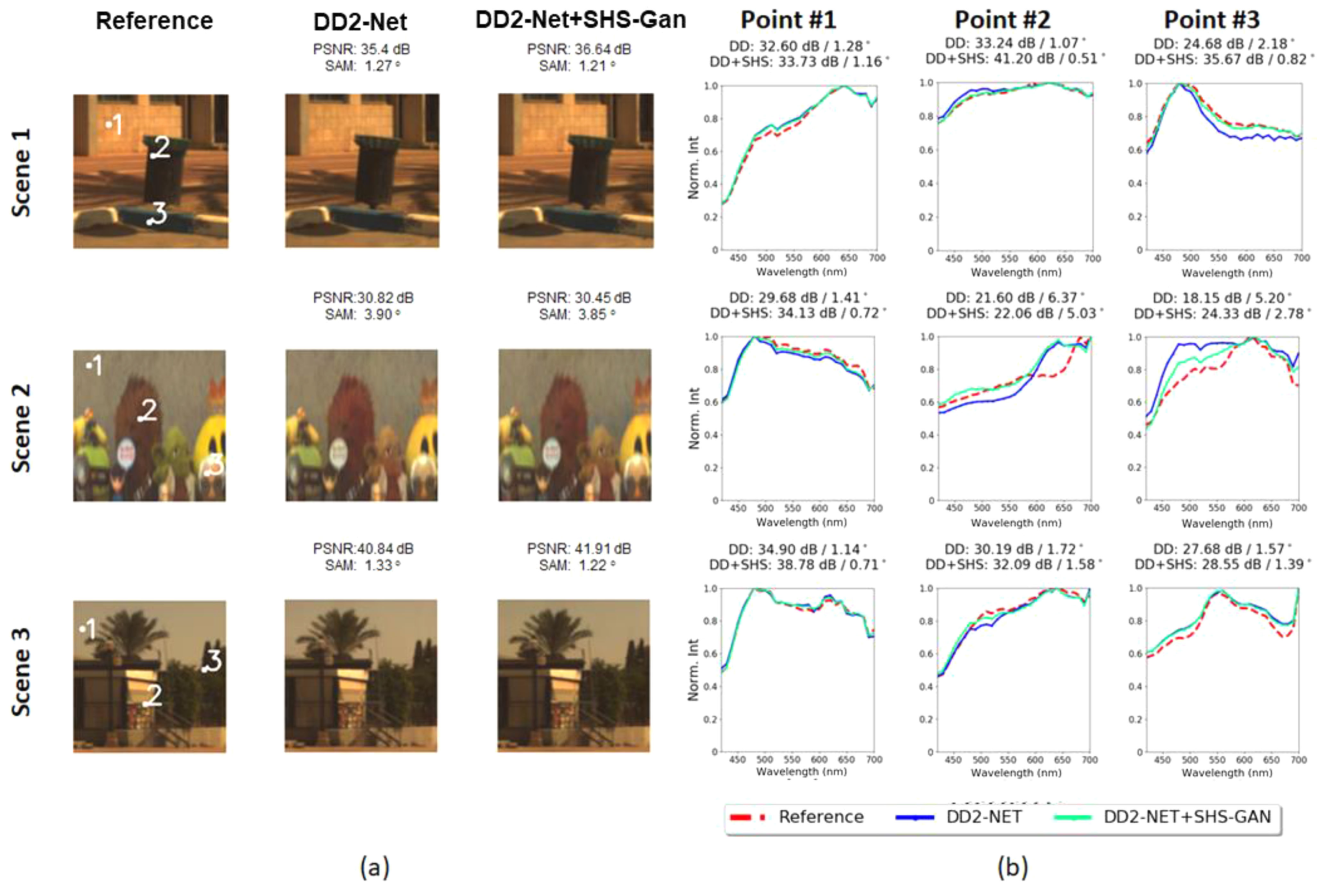

Fig. 10. Simulation results for 3 exemplary scenes for a system configuration of a monochromatic camera with a 2D diffuser. Reconstruction results were obtained with the DD2-Net and with the DD2-Net+SHS-GAN algorithms. (a) RGB representations of the reference and the reconstructed HS cubes, with the average reconstruction's PSNR and SAM values. (b) Reference (dashed red) and reconstructed spectra achieved by the DD2-Net algorithm (blue) and by the DD2-Net+SHS-GAN algorithm (light green), at selected spatial positions marked in the corresponding reference RGB image to the left.

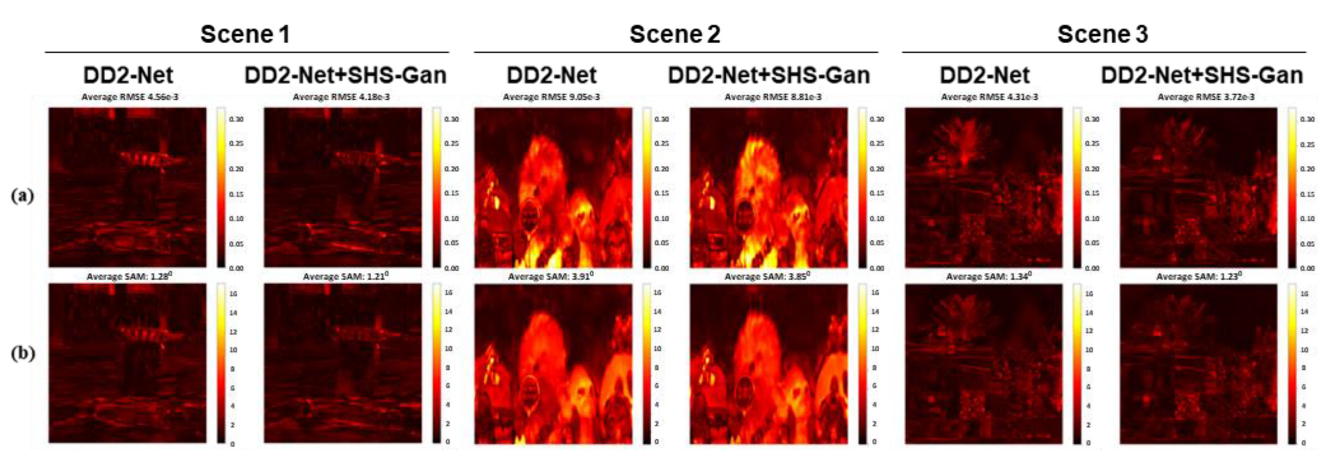

Fig. 11. Error maps for the 3 exemplary scenes for a system configuration of a monochromatic camera with a 2D diffuser. Reconstruction results were obtained with the DD2-Net and with the DD2-Net+SHS-GAN algorithms. (a) RMSE error maps. (b) SAM error maps.

TABLE II
LAB EXPERIMENTAL RECONSTRUCTION RESULTS SUMMARY FOR 31 TEST CUBES

Training	**PSNR [DB]**			**SSIM**			**SAM [DEG]**		
ALGORITHM	BEST	WORST	MEAN	BEST	WORST	MEAN	BEST	WORST	MEAN
DD1-NET	33.021	**17.294**	25.114	0.980	0.906	0.952	5.429	23.431	10.704
DD1-NET + SHS-GAN	**34.509**	15.599	**25.466**	**0.983**	**0.924**	**0.958**	**4.929**	**21.953**	**9.764**

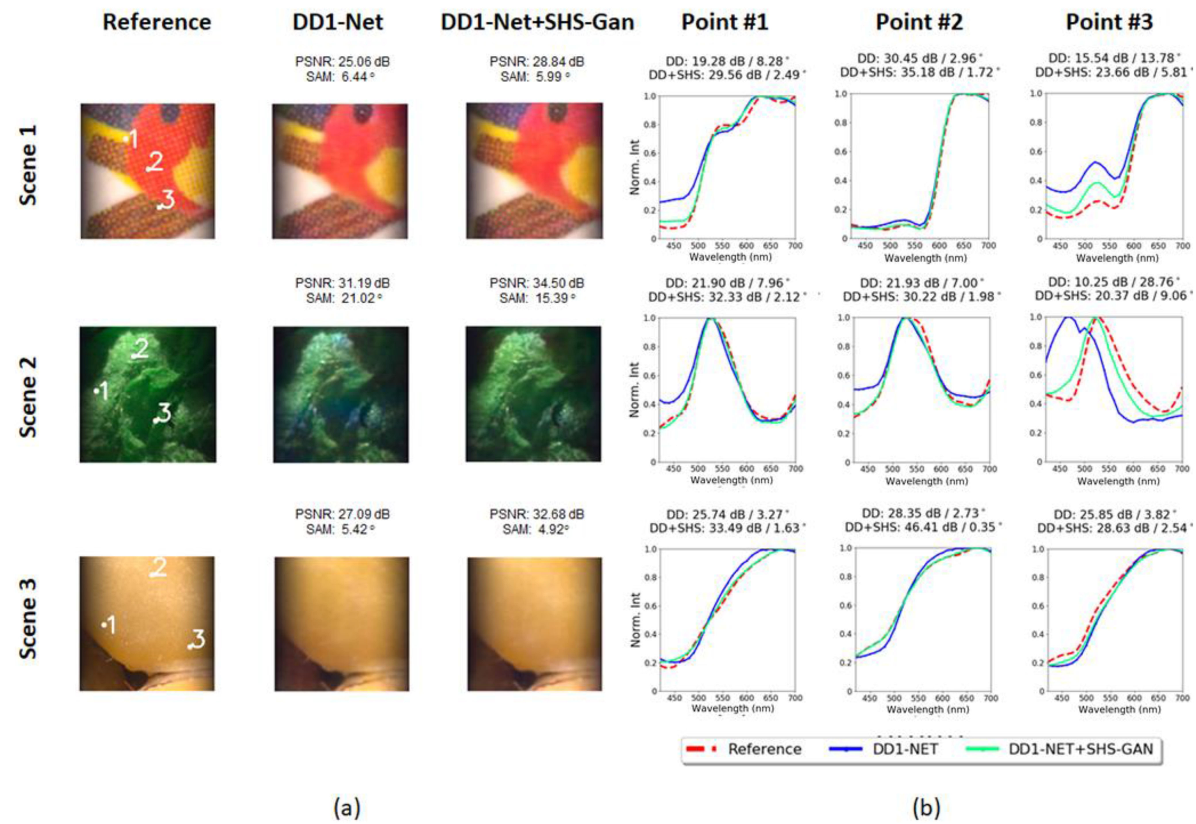

Fig. 12. Lab experimental reconstruction results for 3 exemplary scenes of real, reflective objects which were obtained by a monochromatic camera with a 1D diffuser. Reconstruction results were obtained with the DD1-Net and with the DD1-Net+SHS-GAN algorithms. (a) RGB representations of the reference and the reconstructed HS cubes, with the average reconstruction's PSNR and SAM values. (b) Reference (dashed red) and reconstructed spectra achieved by the DD1-Net algorithm (blue) and by the DD1-Net+SHS-GAN algorithm (light green), at selected spatial positions marked in the corresponding reference RGB image to the left.

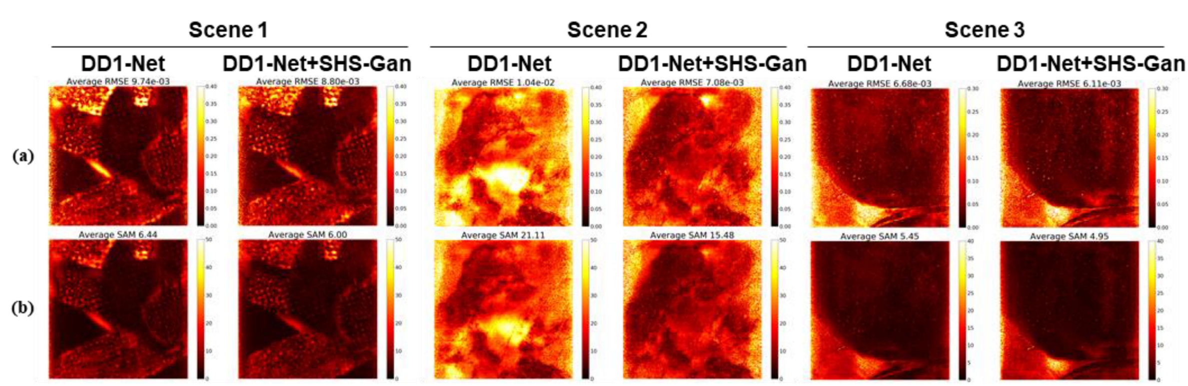

Fig. 13. Error maps for 3 exemplary scenes of real, reflective objects which were obtained by a monochromatic camera with a 1D diffuser. (a) RMSE error maps. (b) SAM error maps.

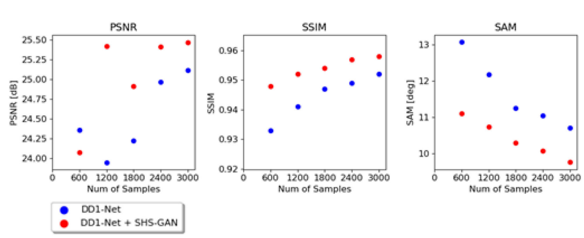

Fig 14. Average reconstruction result of the PSNR, SSIM, and SAM metrics versus the number of real samples.

as the SSIM figure of merit (+0.015). In the remaining runs, an improvement was achieved in all the figures of merit when enriching the data with synthetic HS cubes. While an expected trendline of reconstruction improvement is present as the number of natural HS cubes increases, an interesting observation is a monotonic decrease in the improvement of each figure of merit. This behavior can be explained by the fact that the number of synthetic HS cubes was constant (20000) for each number of natural HS cubes used in the training phase. Hence, the natural HS cubes have a gradually stronger impact on the reconstruction

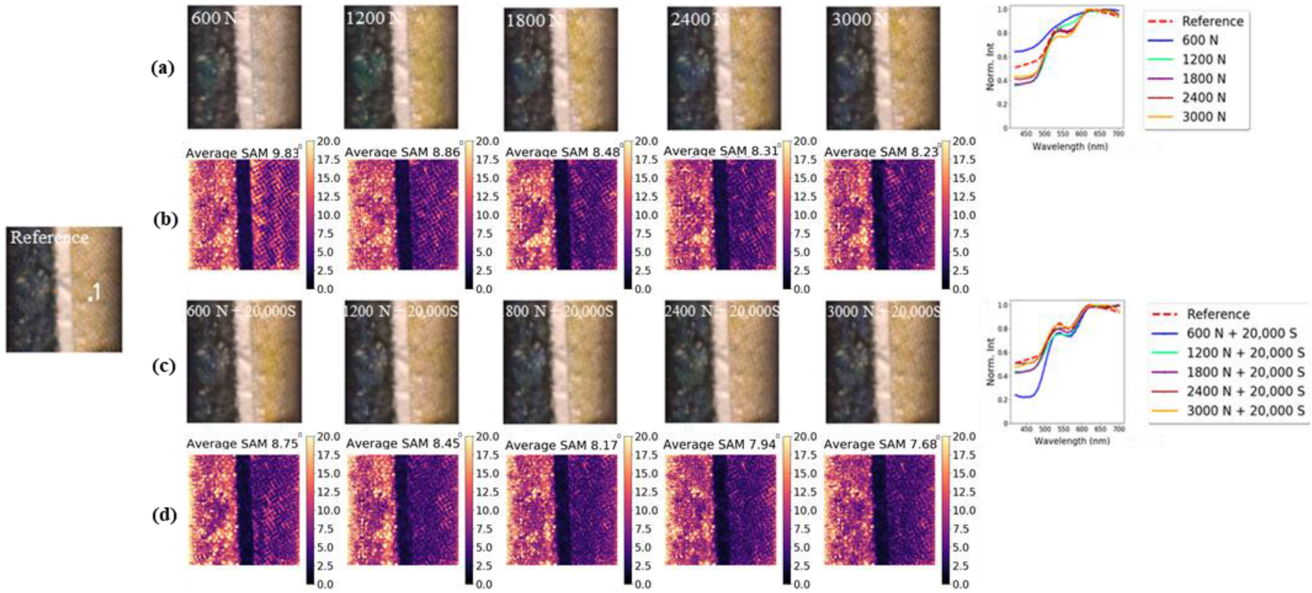

Fig. 15. Reconstruction results for a HS cube of a real paper sticker object with different number of real HS samples used in the training stage. (a) DD1-Net algorithm; (b) DD1-Net + SHS-GAN algorithm. The first column from the left is the RGB representation of the reference HS cube. The remaining columns show the RGB representations of the reconstructed HS cubes for the specified number of samples, with average PSNR and SAM values and SAM error maps. The rightmost column presents the spectra at point '1' as marked on the reference RGB image for Reference spectrum (Dashed red), and the reconstructed spectra for each number of natural HS cubes.

quality as their number rises. Fig. 15 shows an example for the quality of the reconstructions of a paper sticker object for each number of natural HS cubes and for each reconstruction method (DD1-Net and DD1-Net + SHS-GAN), including the corresponding SAM error maps.

VII. Discussion and Conclusion

In this work, we presented a method that utilizes a WGAN framework for enriching HS databases in the visible wavelength domain. Based on spectral statistics from a limited amount of natural HS cubes, our framework can expand an unlimited-sized RGB database into a synthetic HS database.

To generate data that contains natural statistics in both spectral and spatial domains, we integrated FFT-based data processing and RGB-based conditional regularization. To the best of our knowledge, this is a novelty of our framework.

To evaluate the contribution of the synthetic HS data on common DL-based, HS imaging tasks, we integrated the SHS-GAN framework for the task of snapshot spectral imaging using our recently published DD-Net algorithm [10]. This integration was demonstrated to be promising and improved the reconstruction quality in both simulations and lab experiments of HS reconstruction, for natural outdoor scenes and various miniature reflective objects, respectively. In addition, we demonstrated that the contribution of the synthetic data becomes more significant as the size of the natural HS database is smaller (assuming that the HS data still represents the desired task). This observation makes our suggested approach particularly appealing for cases where only a small number of natural HS cubes available for the NN training, as often happens in HS tasks. Despite its promising potential, our method also has a few limitations to consider and address in future research: First, the RGB images mainly affect the spatial characteristics of the synthetic data, while the spectral characteristics are learned from the natural HS examples distribution. Therefore, the SHS-GAN enriches the spatial characteristics but still cannot generate spectral characteristics that do not appear in some manner in the HS training database. Second, the suggested framework is currently suitable only for HS cubes in the visible range. However, we believe that it could be generalized to other regions of the optical spectrum, by using conditional inputs such as grayscale images or multispectral cubes with limited number of spectral channels. Finally, in the context of HS reconstruction, one still needs a reliable model of the optical system (e.g., a precise characterization of the point spread function) to simulate the corresponding system response.

We believe that the results of this work can provide an efficient framework for improving the quality of other data-driven tasks in the field of Computational Photography, and especially for tasks of computational spectral imaging and snapshot spectral imaging.

References

[1] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Neural Inf. Process. Syst., NIPS*, 2012, pp. 1097–1105.

[2] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," 2020, *arXiv:2001.05566*. [Online]. Available: http://arxiv.org/abs/2001.05566

[3] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.,* 2015, pp. 5162–5170.
[4] L. X. Xie, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Proc. Adv. Neural Inform. Proc. Syst.,* Lake Tahoe, NV, USA, 2012, pp. 350–358.
[5] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.
[6] Q. Xie, M. Zhou, Q. Zhao, D. Meng, W. Zuo, and Z. Xu, "Multispectral and hyperspectral image fusion by MS/HS fusion net," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.,* 2019, pp. 1585–1594.
[7] I. Choi, D. S. Jeon, G. Nam, D. Gutierrez, and M. H. Kim, "High-quality hyperspectral reconstruction using a spectral prior," *ACM Trans. Graph.*, vol. 36, no. 6, p. 218, 2017.
[8] L. Wang, C. Sun, Y. Fu, M. H. Kim, and H. Huang, "Hyperspectral image reconstruction using a deep spatial-spectral prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.,* Jun. 2019, pp. 8024–8033.
[9] X. Miao, X. Yuan, Y. Pu, and V. Athitsos, "λ-net: Reconstruct hyperspectral images from a snapshot measurement," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 4059–4069.
[10] J. Hauser, A. Zeligman, A. Averbuch, V. A. Zheludev, and M. Nathan, "DD-Net: Spectral imaging from a monochromatic dispersed and diffused snapshot,". *App. Opt.*, vol. 59, pp. 11196–11208, 2020.
[11] D. S. Jeon *et al.*, "Compact snapshot hyperspectral imaging with diffracted rotation," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–14, 2019.
[12] M. E. Toivonen, C. Rajani, and A. Klami. "Snapshot hyperspectral imaging using wide dilation networks," *Mach. Vis. Appl.*, vol. 32, no. 9, 2021.
[13] B. Arad, and O. Ben-Shahar, "Sparse recovery of hyperspectral signal from natural RGB images," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 19–34.
[14] A. Chakrabarti, and T. Zickler, "Statistics of real-world hyperspectral images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 193–200.
[15] G. Vane, R. Green, T. Chrien, H. Enmark, E. Hansen, and W. Porter, "The airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sens. Environ.*, vol. 44, no. 2/3, pp. 127–143, May/Jun. 1993.
[16] D. Foster, K. Amano, S. Nascimento, and M. Foster, "Frequency of metamerism in natural scenes," *J. Opt. Soc. Amer. A*, vol. 23, no. 10, pp. 2359–2372, 2006.
[17] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based data augmentation for improved liver lesion classification," in *Proc. 1st Conf. Med. Imag. Deep Learn*., 2018, pp. 1–3.
[18] S. Tripathi, S. Chandra, A. Agrawal, A. Tyagi, J. M. Rehg, and V. Chari, "Learning to generate synthetic data via compositing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.,* 2019, pp. 461–470.
[19] N. Audebert, B. L. Saux, and S. Lefèvre, "Generative adversarial networks for realistic synthesis of hyperspectral samples," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2018, pp. 5091–5094.
[20] Y. Lu, D. Perez, M. Dao, C. Kwan, and J. Li, "Deep learning with synthetic hyperspectral images for improved soil detection in multispectral imagery," *IEEE Ubiquitous Comput., Electron. Mobile Commun. Conf.*, New York, NY, USA City, pp. 117–140, 2018.
[21] K. Jiang, W. Xie, Y. Li, J. Lei, G. He, and Q. Du, "Semisupervised spectral learning with generative adversarial network for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 5224–5236, Jul. 2020.
[22] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[23] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 1125–1134.
[24] X. Yan, J. Yang, K. Sohn, and H. Lee, "Attribute2Image: Conditional image generation from visual attributes," in *Proc. Eur. Conf. Comput. Vis.,* 2016, pp. 776–791.
[25] T. Xu *et al.*, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," 2017, *arXiv:1711.10485*.
[26] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[27] W. Zhao, X. Chen, J. Chen, and Y. Qu, "Sample generation with self attention generative adversarial adaptation network (SaGAAN) for hyperspectral image classification," *Remote Sens*, vol. 12, no. 5, pp. 7354–7363, 2020.
[28] J.-C. Burnel, K. Fatras, and N. Courty. "Generating natural adversarial hyperspectral examples with a modified wasserstein GAN," 2020. [Online]. Available: http://arxiv.org/abs/2001.09993
[29] J. Feng, H. Yu, L. Wang, X. Cao, X. Zhang, and L. Jiao, "Classification of hyperspectral images based on multiclass spatial-spectral generative adversarial networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5329–5343, Aug. 2019.
[30] X. Liu, Y. Qiao, Y. Xiong, Z. Cai, and P. Liu, "Cascade conditional generative adversarial nets for spatial-spectral hyperspectral sample generation," *Sci. China Inf. Sci.*, vol. 63, no. 4, pp. 1–140, Apr. 2020.
[31] K. G. Lore, K. K. Reddy, M. Giering, and E. A. Bernal, "Generative adversarial networks for spectral super-resolution and bidirectional RGB-ToMultispectral mapping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 926–933.
[32] J. Hauser *et al.*, "OMHS - The objects mosaic hyperspectral database," 5 May, 2021. [Online]. Available: https://ieee-dataport.org/documents/omhs-objects-mosaic-hyperspectral-database
[33] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," In *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 214–223.
[34] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *in Proc. Int. Conf. Learn. Represent. (ICLR),* 2018, pp. 1–26.
[35] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3722–3731.
[36] Y. Taigman, A. Polyak, and L. Wolf. "Unsupervised cross-domain image generation," in *Proc. Internat. Conf. Learn.*, 2017, pp. 1–14.
[37] D.H Foster and K. Amano, "Hyperspectral imaging in color vision research: tutorial," *J. Opt. Soc. Ameri.*, 2019, 36, 606-627.
[38] X. Wang, and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 318–335.
[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 600–612, Apr. 2004.
[40] D. P. Kingma, and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent*., 2015, pp. 1–41.
[41] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning—Lecture 6a: Overview of mini-batch gradient descent," Univ. Toronto, Toronto, ON, Canada, 2012. Mar. 2021. [Online]. Available: https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf
[42] W. Geisler, and J. Perry, "Statistics for optimal point prediction in natural images," *J. Vis.*, vol. 11, no. 12, pp. 1–17, 2011. Mar. 2021 [Online]. Available: http://natural-scenes.cps.utexas.edu/db.shtml
[43] J. Burge, and W. S. Geisler. "Optimal defocus estimation in individual natural images," in *Proc. Nat. Acad. Sci.*, 2011, pp. 16849–16854.
[44] A. Savitzky, and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Anal. Chem.*, vol. 36, pp. 1627–1639, 1964.

Jonathan Hauser received the B.Sc. and M.Sc. degrees in electrical engineering from the Faculty of Engineering, Tel Aviv University, Tel Aviv, Israel, in 2011 and 2016, respectively, under the supervision of Prof. Menachem Nathan and the late Prof. Michael A. Golub. He is towards the completion of the Ph.D. degree in electrical engineering in Tel Aviv University, under the supervision of Prof. Menachem Nathan and Prof. Amir Averbuch. His research interests include optics, image processing, and computational imaging, with emphasis on computational snapshot spectral imaging.

Gal Shtendel received the B.Sc. degree in electrical engineering from the Technion—Israel Institute of Technology, Haifa, Israel, in 2018. He is currently working toward the M.Sc. degree in electrical engineering with Tel Aviv University, under the supervision of Prof. A. Averbuch and Prof. M. Nathan. His research interests include signal processing and visual data analysis, computational imaging, and deep learning.

Amit Zeligman received the B.Sc. degree in electrical engineering from the Ben-Gurion University of the Negev, Beersheba, Israel, in 2014, and the M.Sc. degree in electrical engineering from the Faculty of Engineering, Tel Aviv University, Tel Aviv, Israel, in 2021, under the supervision of Prof. A. Averbuch and M. Nathan.

Amir Averbuch was born in Tel Aviv, Israel. He received the B.Sc. and M.Sc. degrees in mathematics from the Hebrew University in Jerusalem, Israel, in 1971 and 1975, respectively. He received the Ph.D degree in computer science from Columbia University, New York, in 1983. During 1966–1970 and 1973–1976, he served in the Israeli Defense Forces. In 1976–1986, he was a Research Staff Member with IBM T.J. Watson Research Center, Yorktown Heights, in NY, Department of Computer Science. In 1987, he joined the School of Mathematical Sciences (later split to School of Computer Science), Tel Aviv University, where he is currently Professor of Computer Science. His research interests include applied & computational harmonic analysis, big data processing and analysis, signal/image processing, numerical computation and scientific computing (fast algorithms). In 2013, he founded ThetaRay with Prof. Ronald Coifman from Yale University.

Menachem Nathan received the Ph.D. degree from Cornell University, Ithaca, NY, in 1979. He is Professor Emeritus, School of Electrical Engineering, Tel Aviv University, Israel. Before joining Tel Aviv University, he held a Postdoctoral position with the IBM Research Labs, Yorktown Heights, NY (1979–1981), and was a Senior Scientist with Martin Marietta Laboratories in Baltimore, MD (1981–1988). He has published about 110 papers in refereed journals, and holds 40 patents in a wide variety of subjects.