

Dissertation for Doctor of Philosophy

Computational Spectrometers Based on
Multilayer Thin Films

Cheolsun Kim

School of Electrical Engineering and Computer Science

Gwangju Institute of Science and Technology

2022

박 사 학 위 논 문

다층 박막 기반 계산 분광기

김 철 순

전 기 전 자 컴 퓨 터 공 학 부

광 주 과 학 기 술 원

2022

Computational Spectrometers Based on Multilayer Thin Films

Advisor: Professor Heung-No Lee

by

Cheolsun Kim

School of Electrical Engineering and Computer Science
Gwangju Institute of Science and Technology

A dissertation submitted to the faculty of the Gwangju Institute of Science and Technology in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering and Computer Science

Gwangju, Republic of Korea

November 15, 2021

Approved by



Professor Heung-No Lee

Committee Chair

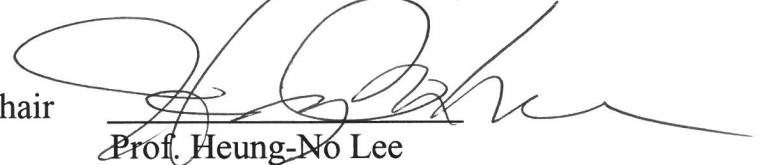
Computational Spectrometers Based on Multilayer Thin Films

Cheolsun Kim

Accepted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

November 15, 2021

Committee Chair


Prof. Heung-No Lee

Committee Member


Prof. Boreom Lee

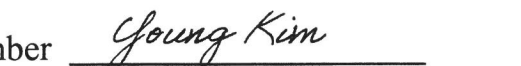
Committee Member


Prof. Jong Won Shin

Committee Member


Prof. Young Min Song

Committee Member


Prof. Young L. Kim (Purdue Uni.)

Dedicated to my family

Ph.D./EC Cheolsun Kim. Computational Spectrometers Based on Multilayer
20164005 Thin films. Electrical Engineering and Computer Science, 2022. 80p.
Advisor: Prof. Heung-No Lee.

Abstract

A spectrometer is a device that measures the intensity of light emitted or absorbed by an object as a function of wavelength. It is used in various research fields and industries, such as chemical analysis, remote sensing, and color inspection. Recently, spectral technology has attracted researchers' attention because of its potential to improve the quality of life of ordinary citizens. However, spectrometers face several limitations that they are expensive, heavy, large in size, and require sophisticated manipulation. One approach to overcome these limitations is computational spectroscopy using multilayer thin films (MTF). The MTF can be realized to have unique transmission characteristics by controlling the thickness and number of layers and can be made small in size. In this dissertation, we present a computational spectrometer using multilayer thin films

In the first half of this dissertation, we present the design, fabrication, and implementation of an MTF filter array for computational spectroscopy. In general, an MTF filter used in a filter-based spectrometer is designed in a bandpass type that only transmits a specific wavelength. To use a bandpass type filter as a spectrometer, a set of filters that continuously transmits signals of different wavelengths is required, and

plenty of filters are needed to cover a wide wavelength range. In addition, to achieve high resolution, it is necessary to fabricate a sophisticated filter with a narrow transmission width. Unlike band-pass filters, the proposed MTF filters were designed to transmit a signal in unique transmission patterns over a wide wavelength range. The MTF filters were fabricated all at once in the form of an array using a combinatorial deposition technique. By attaching the fabricated filter array to a complementary metal-oxide-semiconductor (CMOS) image sensor, we built the device for computational spectroscopy.

In the second half of this dissertation, we present the reconstruction of unknown spectra using computational approaches. An unknown incident spectrum is modulated by the fabricated MTF filter array and measured by a CMOS image sensor. The measured intensities are signals spectrally modulated by the unique transmissions of the MTF filters. The relation among the unknown incident spectrum, transmissions of MTF filters, and the measured intensities can be expressed in linear equations. Using a small number of linear equations, it is possible to reconstruct the unknown incident spectrum in high resolution from the measured intensities by applying computational approaches. We implement two computational approaches: numerical optimization based on L_1 norm and deep learning based on convolutional neural network. Optical experiments were conducted to demonstrate the reconstruction performances of these two approaches using various kinds of light sources. Finally, the reconstruction performances of these two approaches in the noisy environments were comparatively analyzed through simulations.

©2022

Cheolsun Kim

ALL RIGHTS RESERVED

Ph.D./EC 김철순. 다층 박막 기반 계산 분광기. 전기전자컴퓨터공학,
20164005 2022. 80p. 지도교수: 이흥노.

국 문 요 약

분광기는 어떤 물체가 방출하거나 흡수하는 빛의 세기를 파장에 따라 측정하는 장치이다. 화학분석, 원격탐사, 색상 검사 등 다양한 연구 분야와 산업계에서 분광기는 활용되고 있다. 최근, 분광 기술을 활용하여 민간 삶의 질을 향상 시킬 수 있다는 잠재력으로 인해 연구자들의 관심을 받고 있다. 그러나 현재의 분광기는 값이 비싸고, 무게가 무겁고, 크기가 크며, 정교한 조작 필요와 같은 여러 한계에 직면해 있다. 이러한 한계를 뛰어넘기 위한 하나의 접근 방법으로 다층 박막을 이용한 계산 분광기가 있다. 다층 박막은 박막의 두께와 층수를 조절하여 고유한 투과 특성을 갖도록 구현할 수 있으며 크기를 작게 제작 할 수 있다. 본 학위논문에서는 다층 박막 기반의 계산 분광기를 제안한다.

본 논문의 전반부에서는 계산 분광기를 위한 다층 박막 필터 배열 설계와 제작 그리고 구현 방법에 대해 논의한다. 일반적으로 필터 기반 분광기에 사용되는 다층 박막은 투과 특성이 특정 파장의 신호만 통과되는 밴드패스 형태로 설계된다. 따라서 밴드패스 형태의 필터를 분광기로 사용하기 위해서는, 연속적으로 서로 다른 파장의 신호를 투과하는 필터 배열이 필요하며 넓은 파장 범위를 다루기 위해서는 무수한 필터가 필요하다. 또한, 고해상도를 달성하기 위해서는 좁은 투과 폭을 가진 정교한 필터 제작이 필요하다. 밴드패스 필터와 달리, 넓은 파장대역에서 신호를

다양한 형태로 투과시키는 필터들을 다층 박막을 통해 설계하였다. 설계된 필터들은 조합 증착 기술을 사용하여 배열 형태로 한번에 제작되었다. 제작된 필터 배열을 CMOS 이미지 센서에 부착하여 계산 분광을 위한 장치를 구축하였다.

본 논문의 후반부에서는 제작된 분광기를 통한 분광 신호 획득과 디지털 신호처리를 활용한 고해상도 분광 신호 복원에 대해 논의 한다. 입력 광원은 제작된 다층 박막 필터 배열에 의해 변조되고 CMOS 이미지 센서에 의해 측정된다. 측정된 신호는 다층 박막 필터의 고유한 투과 특성에 의해 변조된 신호이다. 미지의 입력 스펙트럼, 박막 필터의 투과 특성, 측정 신호 사이의 관계는 선형 관계로 표현된다. 적은 수의 선형 관계들의 조합을 활용하고 디지털 신호처리를 적용하여 측정된 신호에서 미지의 입력 스펙트럼을 고해상도로 재구성 할 수 있다. L_1 놈 기반의 수치최적화와 합성곱 신경망 기반의 딥러닝, 두 가지 디지털 신호처리 접근 방식을 구현하였다. 다양한 종류의 광원을 사용하여 이 두가지 접근 방식의 재구성 성능을 입증할 광학 실험을 수행하였다. 마지막으로 시뮬레이션을 통하여 잡음이 많은 환경에서 이 두가지 접근 방식의 재구성 성능을 비교 분석하였다.

©2022

김 철 순

ALL RIGHTS RESERVED

List of Contents

Abstract	i
List of Contents	v
List of Tables	vii
List of Figures	viii
Chapter 1 Introduction	1
1.1. Research Background	1
1.1.1. Optical Spectrometer	1
1.1.2. Compressed Sensing	2
1.1.3. Deep Learning.....	3
1.2. Outline of this dissertation	3
Chapter 2 Computational Spectrometers	5
2.1. Introduction.....	5
2.1.1. Related Works.....	5
2.1.2. Contributions of this Dissertation	6
2.2. Principle of Computational Spectrometers	7
2.2.1. Conventional Spectrometers	7
2.2.2. Computational Spectrometers	8
2.2.3. Working Principle of Computational Spectrometers	9
Chapter 3 Multilayer Thin films (MTF)-based Computational Spectrometer	11
3.1. MTF Filter.....	11
3.1.1. MTF Filter Design	13
3.1.2. MTF Filter Fabrication	14
3.2. Implementation	19
3.2.1. Implementation of MTF-based Computational Spectrometer	19
3.2.2. Optical Setups for MTF-based Computational Spectrometer.....	19
3.2.3. Characteristics of MTF-based Computational Spectrometer.....	21
Chapter 4 Spectral Reconstruction	24
4.1. Spectral Reconstruction using Numerical Optimization.....	24

4.1.1. Sparse Recovery	24
4.1.2. Optical Experiments.....	24
4.1.3. Pinhole Imaging.....	34
4.1.4. Summary	36
4.2. Spectral Reconstruction using Deep Learning.....	37
4.2.1. Deep Learning Architecture for Spectral Reconstructions	37
4.2.2. Experimental Dataset.....	41
4.2.3. Results.....	42
4.2.4. Summary	48
4.3. Comparison between Numerical Optimization and Deep Learning	49
4.3.1. Motivation.....	49
4.3.2. Deep Learning Architecture.....	50
4.3.3. Experimental Dataset.....	53
4.3.4. Dictionary Learning for Sparse Recovery	57
4.3.5. Results.....	58
4.3.6. Summary	67
Chapter 5 Conclusions.....	69
Bibliography	71
Curriculum Vitae	77
Acknowledgement	79

List of Tables

Table 3.1 Recursion for calculating reflection coefficients.....	12
Table 3.2 Comparison between the fabricated two MTF filter arrays.....	18
Table 4.1 Evaluation of monochromatic lights reconstructions using the Gaussian curve fitting	32
Table 4.2 INFONET Colors dataset.	41
Table 4.3 Description of the spectral datasets	56
Table 4.4 Avg. RMSEs and PSNRs over synthetic datasets.....	60
Table 4.5 Avg. RMSEs and PSNRs for the measured datasets.....	63
Table 4. 6 Avg.RMSE and PSNR under various SNR (dB) with synthetic datasets	66

List of Figures

Figure 2.1 Schematics of optical spectrometers: (a) Grating-based spectrometer, (b) filter array-based spectrometer. (c) filter array-based computational spectrometer. 8

Figure 2.2 Illustration of the working principle of the computational spectrometer. 10

Figure 3.1(a) Schematic of the MTF filter array. (b) Example of two transmission functions for MTF filters. 13

Figure 3.2 (a) Schematic of the first MTF filter array fabrication process. (b) Photograph of the fabricated first MTF filter array. (c) Monochrome image of the MTF filter array taken at the wavelength of 700 nm. 15

Figure 3.3 Fabricated the second MTF filter array. (a) 169 identical MTF filter arrays fabricated in a single wafer. (b) Monochrome image of the fabricated MTF filter array illuminated by a halogen light source. 17

Figure 3.4 MTF filter array-based computational spectrometer. (a) Photograph of the MTF filter array, which is directly attached to the CMOS image sensor. (b) Photograph of the CMOS image camera with the fabricated MTF filter array. 19

Figure 3.5 (a) Schematic of the optical setup for measuring transmission functions. (b) Schematic of the optical setup for testing the performance of the proposed spectroscopy system. (c) Photographs of the optical setup and the CMOS image camera with the thin-film filter-array. 20

Figure 3.6 The experimental setup for measuring the transmission functions of the fabricated MTF filter array. 21

Figure 3.7 (a) Two transmission functions of MTF filters. (b) Spectral response of the CMOS image sensor. (c) Spectral sensitivity of an MTF filter with the CMOS image sensor, which can be calibrated by element-wise multiplication of the transmission function and the spectral response of the CMOS image sensor. 22

Figure 3. 8 Heatmap of the sensing matrix. Each row represents the spectral sensitivity with respect to wavelength. (b) Upper triangular matrix of correlation coefficients which are pairwise compared among rows of the sensing matrix. 23

Figure 4.1 (a) Computational reconstruction performance of the fabricated thin-film filter-array with respect to the FWHM. (b) Computational spectral reconstruction performance of the fabricated thin-film filter-array with respect to the SNR. 26

Figure 4.2 Spectral reconstructions of several different input light sources. (a) Spectral reconstructions of monochromatic lights (dots) compared with reference spectra (solid lines): 600 nm (green), 700 nm (yellow), 800 nm (red), and 900 nm (purple). (b) Spectral reconstructions of LEDs (dots) compared with reference spectra (solid lines): green LED (527 nm), and red LED (635 nm). 27

Figure 4.3 Computational spectral reconstruction of a halogen lamp (red dash line) compared with the reference spectrum (black solid line) measured by a conventional spectrometer. . 28

Figure 4.4 Simulation result of two-peaks spectra reconstructions using MTF filter array. (a) An example of two peaks spectra with an FWHM of 1 nm with 2 nm apart. (b) Reconstruction performances of two peaks spectra with respect to signal-to-noise ratios. 29

Figure 4.5 Spectral reconstruction of monochromatic light sources. Monochromatic light with an FWHM of 4 nm at peak wavelengths of (a) 510 nm, (b) 600 nm, (c) 650 nm, (d) 700 nm, (e) 750 nm, and (f) 840 nm. Black solid lines represent reference spectra which are measured by the grating-based spectrometer. Blue circles represent reconstructed spectra using the MTF filter array spectrometer. Red solid lines represent the results of Gaussian fitting. Light gray solid lines represent error between reconstructed and reference spectra. 31

Figure 4.6 Spectral reconstructions of LEDs and a halogen light source. (a) a green LED, (b) an orange LED, (c) a red LED, (d) an infrared LED, (e) Combined two LEDs (orange and red), and (f) a halogen light source. Black solid lines represent reference spectra which are measured by the grating-based spectrometer. Colored circles represent reconstructed spectra using the MTF filter array spectrometer. Light gray solid lines represent error between reconstructed and reference spectra. 33

Figure 4.7 Computational pinhole spectral imaging. (a) Schematic of pinhole imaging; The MTF filter array is placed in front of the pinhole camera. A single filter is adjusted to the pinhole and the filtered image is acquired. By changing filters, 36 filtered images are obtained. (b) 36 filtered images of 8×8 LED matrix showing the number “8”; The upper part consists of green LEDs and the lower part consists of red LEDs. (c) Point-wise spectral reconstruction: a pixel of a green LED block and a red LED block which are denoted in (b).

Light gray solid lines represent error between reconstructed and reference. (d) Monochrome image of reference and reconstructed monochrome images at 571, 600 and 638 nm, respectively. 35

Figure 4.8 DL-based single-shot computational spectrometer. (a) Schematic of the experimental setup. (b) DL architecture, RCNN, comprises convolution blocks, dense blocks, and a residual connection with a learnable sensing matrix. (c) Examples of reconstructed test spectra using a trained RCNN. The black dashed line represents ground truth spectra measured using a commercial spectrometer. The solid green line represents reconstructed spectra using the proposed spectrometer. 39

Figure 4.9 Example of INFONET Colors dataset. (a) monochrome image illuminated by a light source. (b) corresponding spectrum of a light source. 41

Figure 4.10 Result of continuous spectra reconstructions. (a) Histogram of the RMSE distribution for the test dataset; average RMSE is 0.0237. (b) Three examples of the best spectral reconstructions, which are denoted as a green box in the histogram. (c) Three examples of the worst spectral reconstructions, which are denoted as blue boxes in the histogram. The black dashed lines represent GT spectra. The solid colored lines represent reconstructed spectra. 43

Figure 4.11 Results of spectral reconstructions using the normalized and scaled datasets. (a) Test intensities extracted from monochrome images in real experiments are compared with references. The solid red lines represent normalized references for easy comparison with the normalized intensities (solid black lines). The solid gray lines represent scaled intensities. (b) Results of spectral reconstructions from the trained RCNN (solid green lines) using the normalized dataset compared with GT spectra (black dashed lines). (c) Results of spectral reconstructions from the trained RCNN (solid blue lines) using the scaled dataset compared with GT spectra (black dashed lines). The reconstructed spectra of each row in (b) and (c) were obtained using the normalized and scaled test intensities in each row of (a), respectively. 45

Figure 4.12 Reconstructions of transmission spectra of drinks. (a) Photograph of drink samples: Vitaminwater-Power C (pink), Vitaminwater-Energy (yellow), Powerade-Mega Boltz (light yellow), Powerade-Mountain Blast (blue), and Powerade-Purple Strom (purple). (b–f) Reconstructed transmission spectra of drinks. The black dashed lines represent the GT

spectra, and the solid colored lines represent the reconstructed transmission spectra using RCNN. Solid gray lines represent the reconstructed spectra using sparse recovery.	47
Figure 4.13 Overview of computational spectroscopy system including the proposed RCNN: An input spectrum is compressively sampled by the sensing matrix, and the dimension of measurements is extended by the transform matrix. RCNN is trained to recover the original spectrum from the extended measurements.....	51
Figure 4.14 Descriptions of the spectrum recovery process: (a) CNN, (b) RCNN.....	53
Figure 4.15 Examples of two synthetic spectra: the solid purple line represents a Lorentzian line shape spectrum and the solid black line represents a Gaussian line shape spectrum.....	54
Figure 4.16 Sparsifying bases for the sparse recovery: (a) Gaussian line shape matrices and (b) learned dictionaries both from the Gaussian training dataset.	58
Figure 4.17 Spectral reconstructions of test spectra in synthetic datasets, (a) Gaussian dataset, (b) Lorentzian dataset. An input spectrum (solid red (i)) is compared with RCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and ODL (purple (vi)). The baselines are shifted for clarity.	61
Figure 4.18 Spectral reconstructions of test spectra in measured datasets: (a) spectrum of organic compound dibenzothiophene in USGS dataset, (b) spectrum of Munsell color 5PB 2/2. The input spectrum (solid red line (i)) is compared with RCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and K-SVD (purple (vi)). The baselines are shifted for clarity.....	65
Figure 4.19 (a) Mean squared error of Gaussian dataset with respect to epochs. Solid black line denotes validation error of CNN, and solid red line with square symbols denotes validation error of RCNN. (b) Reconstructions of a spectrum with respect to epochs where (i) to (iv) are epochs 1, 50, 150, and 250, respectively. Red line (v) denotes the original spectrum... 68	68

Chapter 1

Introduction

1.1. Research Background

1.1.1. Optical Spectrometer

Optical spectrometers [1] are instruments to measure light intensities with respect to wavelengths. The spectrometers have been used in various fields of industries and researches such as astronomy [2], chemical analysis [3], and remote sensing [4].

Typically, optical Spectrometers can be classified in two ways. First, they can be classified according to the working wavelength range. There are ultraviolet (200 - 400 nm), visible (400 - 700 nm), and infrared (700 nm - 1 mm) spectrometers. Second, they can be classified as structures of the spectrometers. There are grating-based spectrometers and filter-based spectrometers.

Grating-based spectrometers use a diffraction grating, a movable slit, and a photodetector. A diffraction grating disperses a beam passed through the slit into various wavelengths. The detector captures the intensities of dispersed light with respect to wavelengths.

Filter-based spectrometers use one or more optical filters to transmit the selected range of wavelength. As the beam passes through the filter, the desired spectral elements are transmitted. The intensity of the transmitted beam is measured by a photodetector. By changing optical filters, light intensities to wavelengths are measured.

Spectrometers digitize the light intensities and give a plot of intensities with respect to wavelengths.

1.1.2. Compressed Sensing

In the traditional sampling theory, the Nyquist-Shannon sampling theorem, the sampling rate should be at least twice the signal's maximum frequency in order to reconstruct the signal without error. Compressed sensing (CS) [5] is an efficient signal processing framework that requires fewer samples than required by the Nyquist-Shannon sampling theorem to reconstruct the signal by finding a solution to the underdetermined linear system.

There are two conditions for efficient reconstructions in the CS framework. First, the signal should be a sparse signal that a few coefficients are significant and the others are negligible, or sparsely represented in a certain domain, such as Fourier and wavelet. Second, the sensing matrix should have a small mutual coherence [6].

Let us denote the signal as $\mathbf{x} \in \mathbb{R}^{N \times 1}$ and the measurement matrix as $\mathbf{A} \in \mathbb{R}^{M \times N}$, then the sampled signal $\mathbf{y} \in \mathbb{R}^{M \times 1}$ is defined as follows:

$$\mathbf{y} = \mathbf{A}\mathbf{x}. \quad (1)$$

The equation (1) becomes an underdetermined system when we set $M < N$. The underdetermined system has a unique solution if the signal \mathbf{x} is sparse or sparsely represented by a sparsifying basis $\mathbf{\Phi} \in \mathbb{R}^{N \times N}$, i.e., $\mathbf{x} = \mathbf{\Phi}\mathbf{s}$, where $\mathbf{s} \in \mathbb{R}^{N \times 1}$, then the Eq. (1) becomes:

$$\mathbf{y} = \mathbf{A}\mathbf{\Phi}\mathbf{s} = \mathbf{\Psi}\mathbf{s}, \quad (2)$$

where, $\mathbf{\Psi} \in \mathbb{R}^{M \times N}$ is the sensing matrix and \mathbf{s} is a sparse signal. The solution of sparse signal $\hat{\mathbf{s}}$ can be obtained by solving the numerical optimization problem, if the sensing matrix $\mathbf{\Psi}$ satisfy the restricted isometry property (RIP) [7].

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \|\mathbf{s}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - \mathbf{\Psi}\mathbf{s}\|_2 \leq \varepsilon, \quad (3)$$

where ε is a small non-negative constant. Then, the reconstructed signal $\hat{\mathbf{x}}$ is $\mathbf{\Phi}\hat{\mathbf{s}}$.

Over the past decades, researchers have been done a significant effort in applying the CS techniques in many applications such as medical imaging [8], hyperspectral imaging [9], and radar systems [10].

1.1.3. Deep Learning

Deep learning [11] is a part of machine learning, which is based on artificial neural networks. These networks are trained to learn specific tasks, such as recognition and classification from large amounts of datasets. Deep learning has been applied to various fields, such as computer vision, speech recognition, and natural language processing, where they have achieved outperformance over the human being.

Deep learning architectures consist of multiple layers of learnable nodes. Each layer is connected to the previous layer, conducting feature extraction and predictions. After going through the multiple layers, input data becomes the prediction of deep learning architectures. This process is called forward propagation. The input layer is where the input data is fed, and the output layer is where the final prediction or classification is made.

Backpropagation uses gradient descent algorithms to calculate errors between predictions and ground truths. Then, the learnable nodes are adjusted by moving backward through the layers. By conducting the forward propagation and backpropagation, deep learning architectures are learned to improve errors of predictions and classifications.

One of the most popular deep neural networks is convolutional neural networks (CNN) [12]. A CNN automates extracting features from the input data and conducts classification or regressions. This automated feature extraction makes deep learning architectures work well for computer vision applications.

1.2. Outline of this dissertation

In this dissertation, we propose computational spectrometers based on multilayer thin films. We design and fabricate compact spectrometers based on the compressed sensing theory.

Then, computational approaches based on numerical optimization and deep learning are used to reconstruct spectra in a high resolution.

In Chapter 2, we present the working principle of computational spectrometers. We will propose compressed sensing-based optical filters for the spectroscopy system to improve the efficiency of spectrometers. In Chapter 3, we propose multilayer thin films (MTF) for computational spectrometers. In Chapter 4, we demonstrate the MTF-based computational spectrometers using sparse recovery and deep learning approaches. The results of MTF-based computational spectrometers are as follows [13]–[18]:

[13] **Cheolsun Kim**, Woong-Bi Lee, Soo Kyung Lee, Yong Tak Lee, and Heung-No Lee, “Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy,” *Optics and Lasers in Engineering*, Vol. 115, pp. 53-58, Apr. 2019.

[14] **Cheolsun Kim**, Dongju Park, and Heung-No Lee, “Compressive sensing spectroscopy using a residual convolutional neural network”, *MDPI Sensors*, Vol. 20(3), Jan. 2020.

[15] **Cheolsun Kim**, Pavel Ni, Kang Ryeol Lee, and Heung-No Lee, “Mass production-enabled computational spectrometer based on multilayer thin films”, in revision.

[16] **Cheolsun Kim**, Dongju Park, and Heung-No Lee, “Deep learning based single-shot computational spectrometer using multilayer thin films”, in preparation.

[17] Woong-Bi Lee, **Cheolsun Kim**, Gun Wu Ju, Yong Tak Lee, and Heung-No Lee, “Design of thin-film filters for resolution improvements in filter-array based spectrometers using DSP”, *SPIE Defense + Commercial Sensing 2016*, Baltimore, USA, Apr. 17-21, 2016.

[18] **Cheolsun Kim**, Dongju Park, and Heung-No Lee, “Convolutional neural networks for the reconstruction of spectra in compressive sensing spectrometers”, *SPIE Photonics West 2019*, San Francisco, USA, Feb. 2-9, 2019.

Finally, Chapter 5 concludes this dissertation and suggests future research directions.

Chapter 2

Computational Spectrometers

2.1. Introduction

The demand for spectrum information is increasing not only in research and development but also in the private sector. In response to this demand, researchers are trying to make spectrometers that are both small and inexpensive. These spectrometers could be used in various fields, such as medical systems, mobile applications, and remote sensing [4], [19], [20]. In particular, optical filter-based spectrometers do not need motorized or dispersive elements, and their filter array can be directly attached to a complementary metal-oxide-semiconductor (CMOS) image sensor so that they can be easily miniaturized. However, there is a trade-off between size (for integrating filters) and spectral resolution with miniaturized spectrometers. The number of filters that can be integrated into a CMOS image sensor is limited due to its small sensing area. Thus, these spectrometers offer a low spectral resolution.

2.1.1. Related Works

Over the past decade, numerical optimization approaches [21], [22] have been adapted for filter-based spectrometers to improve spectral resolution. The spectral resolution in conventional filter-based spectrometers has been improved using sparse recovery based on numerical optimization [23]. New optical filters have been proposed that work well in numerical optimization approaches and achieve further improvements [24]–[31]. Unlike conventional optical filters, which selectively transmit incident light in specific wavelengths and reflect the remaining wavelengths, these filters, called random spectral filters, modulate and transmit incident light with unique spectral features in the entire wavelength ranges of interest. Various types of random spectral filters have been proposed, such as etalon filters [27], [32], quantum dot filters [26], photonic crystal slabs [29], [33],

and multilayer thin films (MTF) filters [13]. The spectral resolvability of computational spectroscopy has been successfully demonstrated using random spectral filters with low correlation among filters.

Recently, deep learning [11] has been emerging as a promising alternative framework for numerical optimization approaches for computational spectrometers [14], [18], [34]–[37]. Kim, et al. first reported a convolutional neural network (CNN) to restore the spectra from compressively sampled intensities and compared the reconstruction performance between iterative numerical optimization approaches and the CNN via simulation [14], [18]. Real experiments for reconstructing light-emitting diodes (LEDs) using an U-net structure were first reported [34]. Reconstructions of the combination of narrow bands spectra using a programmable supercontinuum laser are demonstrated [36]. Hyperspectral imaging via DL-based filter design and spectra reconstruction was proposed [37].

2.1.2. Contributions of this Dissertation

Our main contribution of this dissertation is that we demonstrate computational spectrometers, which have a low cost, a compact size, a high resolution, a wide working range, and a fast operation time using multilayer thin films. We design multilayer thin films (MTF) filters based on the compressed sensing theory. We propose a mass production enable fabrication process for fabricating MTF filter arrays. We build a spectrometer by attaching the fabricated MTF filter array to a CMOS image sensor.

Using the proposed spectrometer, we measure the intensities of an input light source into a digital signal. From the measured intensities, we reconstruct the spectra of the input light in a high resolution. As the computational approaches, we use sparse recovery based on numerical optimization, and a residual convolutional neural network (RCNN) based on deep learning. We formulate the problem of the underdetermined linear system for spectrometers and solve the problem using these computational approaches.

We conduct numerical experiments for estimating the performances of the computational spectrometer. After numerical experiments, we perform optical experiments using various light sources, including monochromatic lights, LEDs, and a halogen lamp. The results

show good recovery performances over the light sources. In addition, we compare spectral reconstruction performances between sparse recovery and RCNN.

2.2. Principle of Computational Spectrometers

2.2.1. Conventional Spectrometers

Typically, there are two kinds of spectrometers according to optical structures. One is grating-based spectrometers. The other is filter array-based spectrometers.

Grating-based spectrometers use diffractive optics, such as grating and prism, to spread an incident light with respect to wavelengths. Then, the intensities of the spread lights are measured by detectors (CCD or CMOS sensor). The measured intensities change into a digital signal. The spectrometer gives a plot of intensities to wavelengths as shown in Fig 2.1(a). The grating-based spectrometers require diffractive optics and motorized equipment for spreading the incident light. In addition, a long light way is required to achieve enough spreading for a high resolution. Thus, the grating-based spectrometers have a bulky size and a high cost and require a sophisticated operation to use.

On the other hand, filter array-based spectrometers use one or more optical filters to transmit the selected range of wavelengths (bandpass filters). As an incident light passes through the filter, the desired spectral elements are transmitted and the others are reflected. Using a set of filters that transmit different wavelengths and a CMOS image sensor, a filter-based spectrometer could be built. The filtered light intensities from the set of filters are measured by the CMOS image sensor. The measured intensities change into a digital signal. The spectrometer gives a plot of intensities to wavelengths as shown in Fig 2.1(b). The filter-based spectrometers have a compact size and a lightweight. Also, they have a fast acquisition time thanks to their snapshot structure. However, the filter-based spectrometers have a low spectral resolution due to the small number of filters integrated into the CMOS image sensor. In addition, it is difficult to fabricate sophisticated narrow bandpass filters in a uniform spacing.

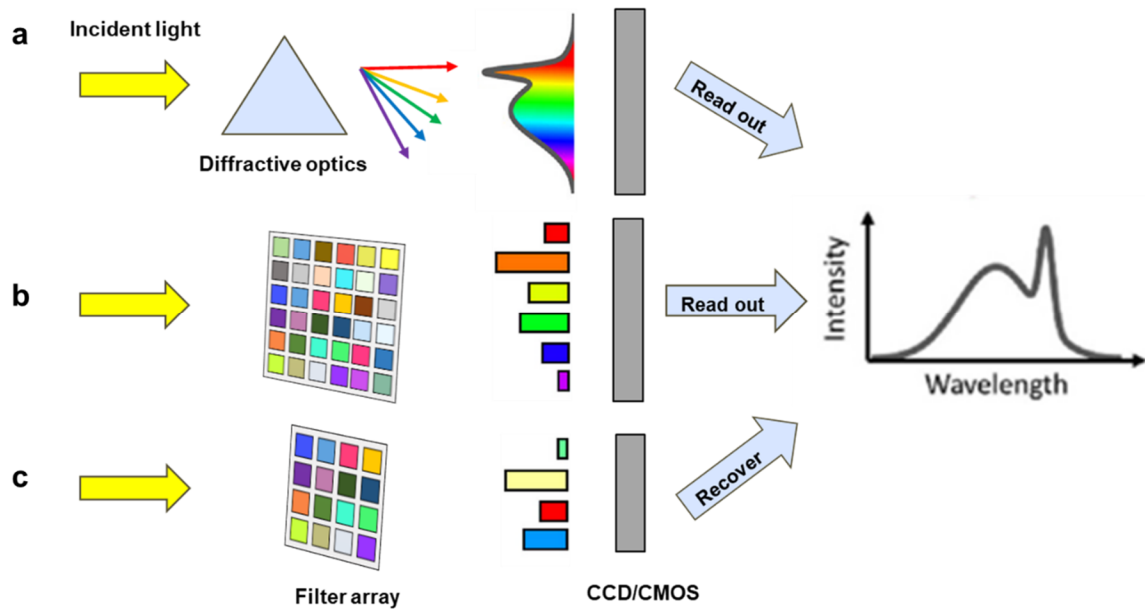


Figure 2.1 Schematics of optical spectrometers [27]: (a) Grating-based spectrometer, (b) filter array-based spectrometer. (c) filter array-based computational spectrometer.

2.2.2. Computational Spectrometers

Figure 2.1(c) shows a schematic of a filter array-based computational spectrometer. The incident light is fed into a filter array and the filter array modulates the incident light in random spectral features. Unlike the bandpass filter that transmits specific wavelengths of the incident light, the filter array in the computational spectrometer transmits incident light with unique spectral features in the entire wavelength ranges of interest. The filter has a transmission function of multiple peaks with various full widths at half maximums (FWHMs) and has a large difference between maximal and minimal transmission in the transmission function. The filtered and modulated light intensities are measured by the CMOS image sensor. The measured intensities change into a digital signal. Using computational approaches, the digital signal could be reconstructed into a high resolution spectrum. Using the filters with random spectral features, we could capture the spectral information of incident light in a wide range of wavelengths without loss. In addition, the computational spectrometers require fewer filters than conventional filter array-based spectrometers.

2.2.3. Working Principle of Computational Spectrometers

As the filter array for computational spectrometer, we use MTF filter array. Let us denote the transmission function of the i -th MTF filter in the wavelength range $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_N]$ as $\mathbf{T}_i = [T_i(\lambda_1), T_i(\lambda_2), \dots, T_i(\lambda_N)]$. the intensity, y_i , measured by CMOS image sensor for an unknown incident spectrum $\mathbf{x} = [x(\lambda_1), x(\lambda_2), \dots, x(\lambda_N)]^T$, can be expressed as:

$$y_i = \sum_{k=1}^N T_i(\lambda_k) Q(\lambda_k) x(\lambda_k) \quad (4)$$

where $\mathbf{Q} = [Q(\lambda_1), Q(\lambda_2), \dots, Q(\lambda_N)]$ is the spectral response of CMOS image sensor in the wavelength range λ . Let us set $R_i(\lambda_k) = T_i(\lambda_k) Q(\lambda_k)$, where $R_i(\lambda_k)$ represents spectral sensitivity of the i -th filter of the CMOS image sensor at the wavelength λ_k , the equation (4) becomes $y_i = \sum_{k=1}^N R_i(\lambda_k) Q(\lambda_k)$. Considering an M number of filters, there is a set of M equations for $i = 1, 2, \dots, M$. The set of M equations can be represented in matrix formation:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} R_1(\lambda_1) & \cdots & R_1(\lambda_N) \\ \vdots & \ddots & \vdots \\ R_M(\lambda_1) & \cdots & R_M(\lambda_N) \end{bmatrix} \begin{bmatrix} x(\lambda_1) \\ \vdots \\ x(\lambda_N) \end{bmatrix} \quad (5)$$

where $\mathbf{y} \in \mathbb{R}^{M \times 1}$ is a column vector with measured intensities from M filters and $\mathbf{R} \in \mathbb{R}^{M \times N}$ is the sensing matrix where each row represents the spectral sensitivity with respect to the wavelength. The spectral sensitivity can be calibrated by element-wise multiplication of the transmission functions of MTF filters and the spectral response of the CMOS image sensor.

Conventional spectrometers read out \mathbf{y} as the incident spectrum \mathbf{x} . In order to make the measured intensities \mathbf{y} as close as possible to the incident spectrum \mathbf{x} , the sensing matrix \mathbf{R} should be an identity matrix with the dimension of $N \times N (M = N)$. This means that N number of filters are needed in conventional manners. In practice, it may be difficult to

fabricate a narrow FWHM filter and, since the number of filters required increases as the wavelength of interest increases, it is more challenging to make a compact spectrometer operating in a wide wavelength range. Unlike conventional filter-based spectrometers, computational spectrometers modulate and measure a wide wavelength range of the incident spectrum using a small number of MTF filters. We consider the sensing matrix \mathbf{R} with dimensions $M \times N (M < N)$. The set of M equations becomes an underdetermined problem. Numerical optimization approaches [38]–[40] and deep learning approaches [35], [41] can be applied to restore the incident spectrum in high resolution by solving the underdetermined problem. Figure 2.2 depicts the working principle of the filter array-based computational spectrometer.

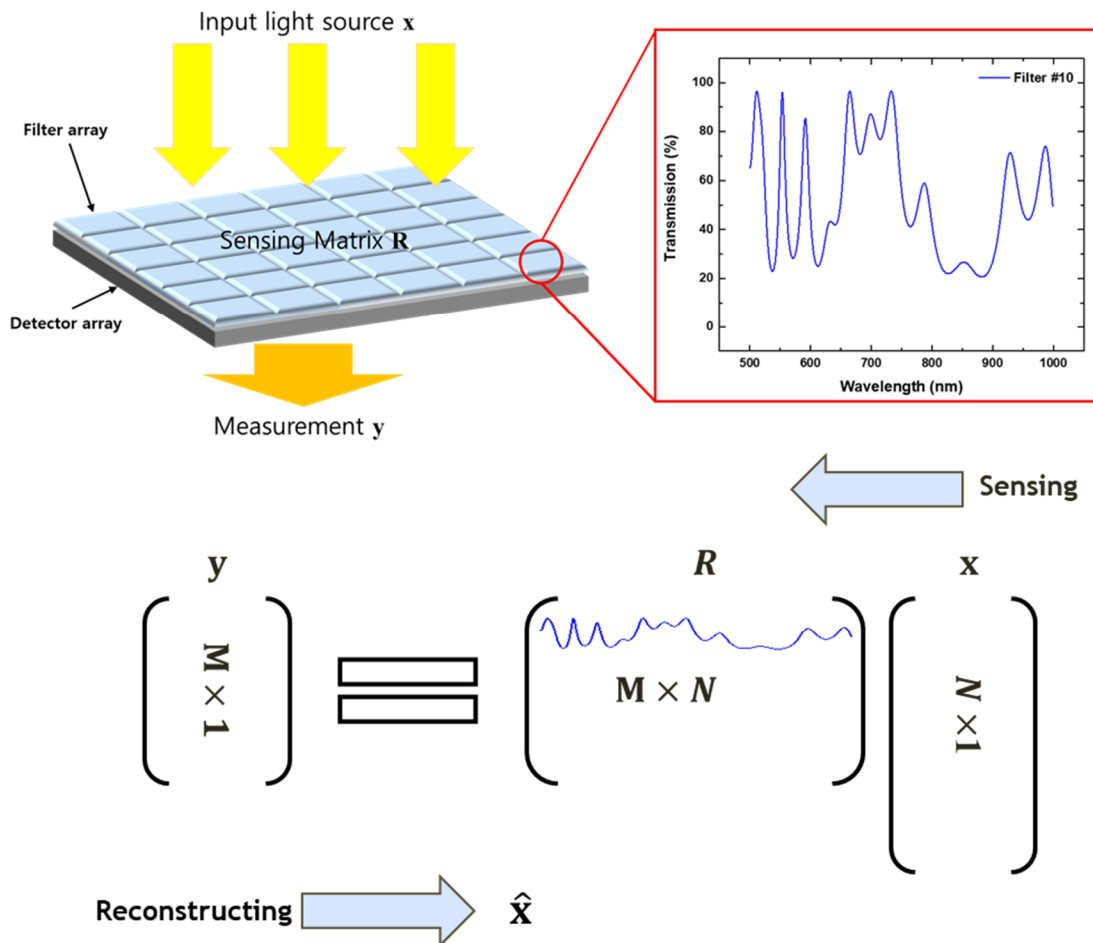


Figure 2.2 Illustration of the working principle of the computational spectrometer.

Chapter 3

Multilayer Thin films (MTF)-based Computational Spectrometer

3.1. MTF Filter

Thin films are basic components that have been applied in a variety of areas, including semiconductor devices, optical coatings, and solar cells [42]. The theoretical transmission function of a multilayer thin-film filter is given by [43]

$$T(\lambda, \theta_1) = 1 - \frac{1}{2} (|\rho_{TE}|^2 + |\rho_{TM}|^2), \quad (6)$$

where ρ_{TE} and ρ_{TM} are the reflection coefficients. Given a wavelength λ and the incident angle θ_1 , transmission function can be calculated using recursive routines shown in Table 3.1

In Table 3.1, given the input of a wavelength λ , a vector of l refractive indices $\mathbf{n} = \{n_1, n_2, \dots, n_l\}$ and a vector of $l-1$ layer thicknesses $\mathbf{d} = \{d_2, d_3, \dots, d_l\}$, a reflection coefficient ρ is generated. Note that there are l layers considered in total. The first one is the layer of the air and the last one is the layer of the substrate. The light is assumed to be arriving from the air to the second layer in normal incidence. The first index n_1 in the vector \mathbf{n} represents the refractive index of the air. The last one n_l in the vector \mathbf{n} represents the refractive index of the substrate. The refractive indices of the intermediate thin-film layers are denoted by n_2 to n_{l-1} . The thickness of the air does not need to be considered. The thickness of the substrate is denoted by d_l .

Table 3.1 Recursion for calculating reflection coefficients.

Input: λ , $\theta_1 = 0$, $\mathbf{n} = \{n_1, n_2, \dots, n_l\}$, $\mathbf{d} = \{d_2, d_3, \dots, d_l\}$.

Step 1: Obtain θ_k , β_k , and N_k

$$\theta_k = \sin^{-1} \left(\frac{n_{k-1}}{n_k} \sin \theta_{k-1} \right), \text{ for } k = 2, 3, \dots, l.$$

$$\beta_k = 2\pi \cos(\theta_k) n_k d_k / \lambda, \text{ for } k = 2, 3, \dots, l$$

$$N_k = \begin{cases} n_k / \cos \theta_k & \text{for } TE \\ n_k \cos \theta_k & \text{for } TM \end{cases}, \text{ for } k = 1, 2, \dots, l$$

Step 2: Set $\eta_l = N_l$

Step 3: Obtain η_2

Decrement k by 1 from $l-1$ to 2

$$\eta_k = N_k \frac{\eta_{k+1} \cos \beta_k + j N_k \sin \beta_k}{N_k \cos \beta_k + j \eta_{k+1} \sin \beta_k}$$

Return η_2

Step 4: Compute $\rho = (N_1 - \eta_2) / (N_1 + \eta_2)$.

Output: ρ

The thicknesses of the intermediate thin-film layers are denoted by d_2 to d_{l-1} . The incidence angle of the light passing from the k th to the $k+1$ st layer is θ_k , and η_k is the effective complex-valued index of the k th layer. A transmission function for a single filter is obtained by considering all wavelengths in the range of interest. An array of transmission functions for the M filters can be obtained by repeating this process where each filter $\mathbf{T}_m \in \mathbb{R}^{1 \times N}$ for $m = 1, 2, \dots, M$ in Eq. (4) is generated from a unique set of refractive index and thickness vectors.

3.1.1. MTF Filter Design

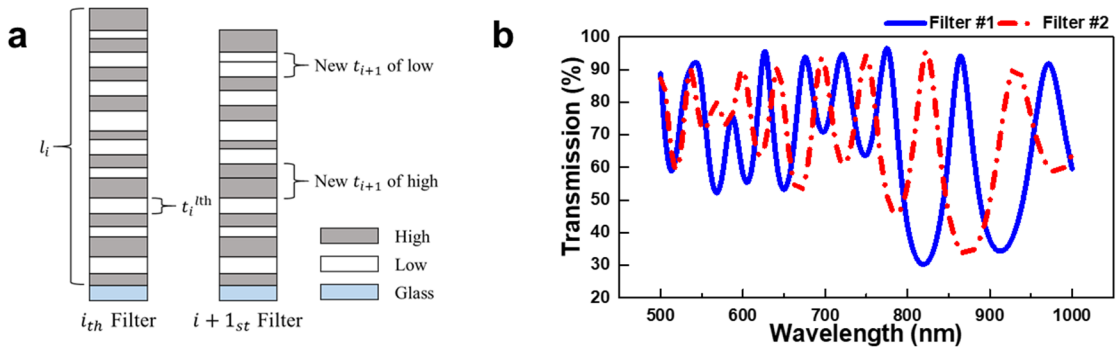


Figure 3.1(a) Schematic of the MTF filter array. (b) Example of two transmission functions for MTF filters.

To implement the proposed 2D filter-array, we numerically modeled the proposed spectroscopy method with reference to [43]–[45], and according to the following steps. (i) Generate the reference vector of layer thicknesses for the reference filter. (ii) Generate a vector of thicknesses for the other filter by randomly removing one to five layer thicknesses from the reference vector. (iii) Repeat the step (ii) 35 times to create a total of 36 vectors of thicknesses. (iv) Use the recursion Table 3.1 and Eq. (6) to calculate the transmission functions for a new filter-array. (v) Use the mutual coherence μ to quantify the goodness of the sensing matrix of the designed filter-array. Mutual coherence is defined as $\mu \triangleq \max_{i,j} |o_{ij}|$, where o_{ij} is the (i, j) th off-diagonal element of the Gram matrix, $\mathbf{T}^* \mathbf{T} \in \mathbb{R}^{N \times N}$. \mathbf{T}^* denotes the conjugate transpose of \mathbf{T} . With these steps, we can generate a single set of 36 filters. By repeating these steps, multiple sets of 36 filters can be obtained.

Among these sets of filter-arrays, the set of filters with a smallest mutual coherence is selected.

In the compressed sensing framework, a sensing matrix with a smaller mutual coherence is better than the one with a higher mutual coherence to capture the information of input signal to be reconstructed [6], [24]. A schematic of the proposed filter-array is shown in Fig. 3.1(a). Each time a layer is removed, the layers above and those below come together to form a single layer with two thicknesses added up. We consider two materials, high- and low-refractive index materials. Figure 3.1(b) shows the transmission functions for two designed filters generated by using two materials, SiN_x and SiO_2 . In conventional spectroscopy, the transmission functions with a large spectral depth and a narrow spectral peak are preferred in order to prevent interference among measurements. In computational spectroscopy, each transmission function of the filter should be wide enough so that the set of the small number of filters fully senses the spectral information in the given wavelength range.

Each filter shows several spectral peaks and rapid changes of transmission value with respect to wavelength. Therefore, each filter has a high optical throughput that the energy (intensity) which passes through the filter is higher than that with the conventional bandpass filter approach. In addition, fewer filters can be used to cover the entire wavelength range with the proposed method. For example, suppose 250 bandpass filters are used to cover the wavelength range from 500 nm to 1000 nm. Then, the bandwidth of the transmission function is 2 nm, according to the conventional bandpass filter design. In the proposed approach, the same range of wavelengths can be covered with only 36 proposed filters.

3.1.2. MTF Filter Fabrication

We fabricated two kinds of MTF filter arrays. The first MTF filter array was fabricated by using combining combinatorial deposition and etching techniques. The second MTF filter array was fabricated by using only the combinatorial deposition technique.

Fabrication of the first MTF filter array. Figure 3.2(a) shows the process in which the first MTF filter array is fabricated. This comprises two main parts; one is SiO_2 film

deposition and the other is SiN_x film deposition according to the specified thicknesses. Prior to depositing an SiO_2 film, a 6×6 germanium (Ge) grid with elements of size $300 \mu\text{m}$ and spacing $100 \mu\text{m}$ was formed on the glass using an e-beam evaporator to separate the filters. In this grid, SiO_2 and SiN_x layers were deposited with the width of $300 \mu\text{m}$ in each filter. Then, selective deposition was done as follow: An intentionally thick SiO_2 film was deposited on the glass patterned with the Ge grid using plasma-enhanced chemical vapor deposition. The regions where the film should not be deposited were then removed by conventional photolithography, namely CF_4/O_2 reactive ion etching. The process pressure and radio frequency power were maintained at 50 mTorr and 50 W, respectively. The SiN_x film deposition process was performed in the same manner as for SiO_2 . Finally, these two main steps, SiO_2 and SiN_x film deposition, were repeated 12 times each to lay down 24 layers. Figures 3.2(b,c) show a photograph of a fabricated MTF filter array and a monochrome image of the filter array, respectively. Each filter is composed of a different number of layers each with different thicknesses; therefore, each one has unique color due to its different transmission function.

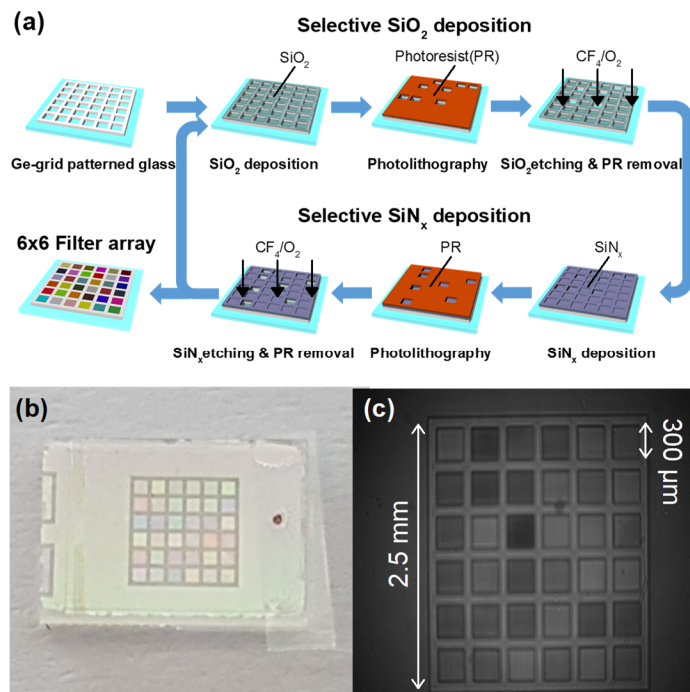


Figure 3.2 (a) Schematic of the first MTF filter array fabrication process. (b) Photograph of the fabricated first MTF filter array. (c) Monochrome image of the MTF filter array taken at the wavelength of 700 nm.

Fabrication of the second MTF filter array. We fabricated 169 identical filter arrays on a single wafer, as shown in Fig. 3.3(a). The filter array is the shape of a 6×6 square grid. The size of the square is $400 \times 400 \mu\text{m}^2$, and the space between the squares is $300 \mu\text{m}$. Accordingly, the size of the filter array is $4.5 \times 4.5 \text{ mm}^2$. To fabricate filter arrays, we used TiO_2 and SiO_2 as high and low refractive index materials, respectively. The refractive indices for TiO_2 and SiO_2 are approximately 2.5 and 1.45, respectively.

Unlike etalon filters that were fabricated by changing the thickness of interspacing dielectric layers [46], [47], we fabricated the MTF filters by changing the number of layers and thicknesses of layers. An MTF filter with a unique transmission function can be obtained by repeatedly alternating the two materials and depositing them with different thicknesses. 36 MTF filters with unique transmission functions were fabricated by selectively omitting certain layers of different MTF filters using shadow masks during the deposition of the MTF filter array. By omitting an intermediate layer, the upper and lower layers sum up to form one layer with a different thickness. The deposition process for creating filter arrays follows.

TiO_2 and SiO_2 films were deposited onto a borosilicate glass wafer whose refractive index is approximately 1.472 at 588 nm. In order to distinguish where the material should be deposited, shadow masks were used. The desired thickness of TiO_2 is deposited on the desired locations using direct current (DC) magnetron sputter. For TiO_2 deposition, a Ti target was sputtered in a mixture of argon (Ar) and oxygen (O_2). The mixture gas flow of 188 sccm of Ar and 12 sccm of O_2 was utilized and the DC power was 700 W. The TiO_2 deposition is performed only on the desired region designated by the shadow mask. Then, the shadow mask is changed, with different patterns on the other mask, and we deposit SiO_2 at the intended thickness. RF magnetron sputter was used for the SiO_2 deposition. A Si target was sputtered in a mixture of Ar and O_2 . The mixture gas flow of 185 sccm of Ar and 15 sccm of O_2 was utilized. The RF power was 300 W. The deposition is repeated 17 additional times by changing the shadow mask and alternating between TiO_2 and SiO_2 . Hence, we conducted ten individual depositions of TiO_2 and nine individual depositions of SiO_2 . After completing thin film deposition, we coated the surface of the thin films with a photoresist. Germanium (Ge) was deposited over the entire wafer area using an e-beam evaporator. Lift-off of the photoresist was performed by soaking the deposited wafer in

acetone. When the photoresist was washed away, Ge deposited on the top of the photoresist was lifted off and washed. After lift-off, a square grid of Ge with the size and spacing of $400\ \mu\text{m}$ and spacing of $300\ \mu\text{m}$ was formed. The wafer cleaning process was then performed, and, finally, the wafer was diced to produce MTF filter arrays. Figure 3.3(b) is a monochrome image of the fabricated MTF filter array illuminated by a halogen light source. The image was taken by the CMOS monochrome camera whose number of pixels is 1280×1024 . As shown in Fig. 3.3(b), we measure uniform intensity using pixels under a single MTF filter. Also, the pixels have unique intensity according to the MTF filter. Using these unique intensities from MTF filters, we can reconstruct the spectrum of unknown incident light. Table 3.2 summarizes the comparison between the fabricated two MTF filter arrays.

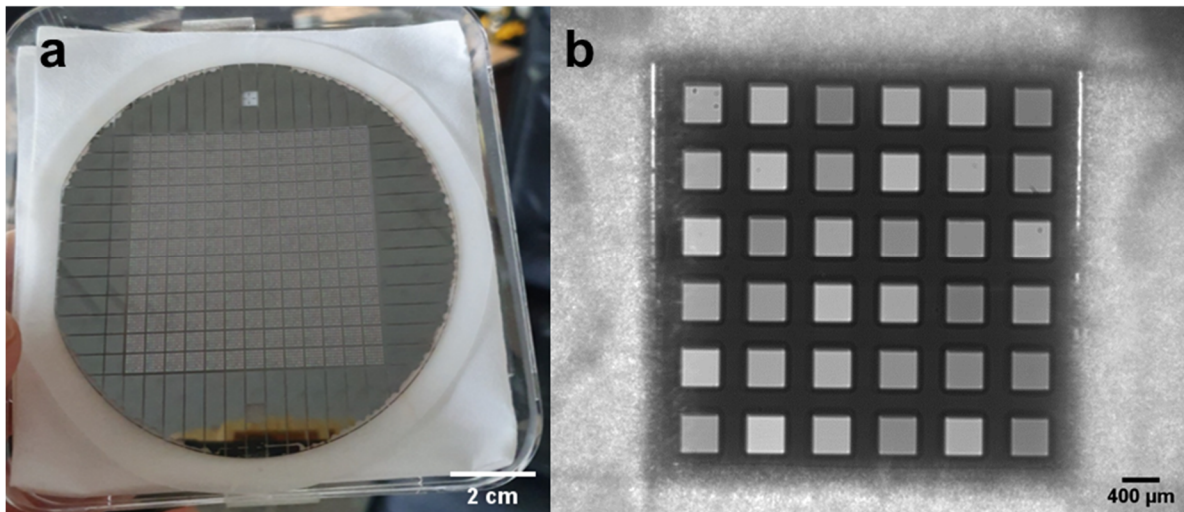


Figure 3.3 Fabricated the second MTF filter array. (a) 169 identical MTF filter arrays fabricated in a single wafer. (b) Monochrome image of the fabricated MTF filter array illuminated by a halogen light source.

Table 3.2 Comparison between the fabricated two MTF filter arrays

	The first MTF filter array	The second MTF filter array
Number of Filters	36	36
Number of Filter arrays	1	169
Materials	SiN _x and SiO ₂	TiO ₂ and SiO ₂
Methods	The combinatorial deposition and etching	The combinatorial deposition
Thickness of layers	100 ~ 300 nm	100 ~ 300 nm
Maximum layers	24	19
Size of filter	300 um × 300 um	400 um × 400 um
Size of filter array	2.5 mm × 2.5 mm	4.5 mm × 4.5 mm

3.2. Implementation

3.2.1. Implementation of MTF-based Computational Spectrometer

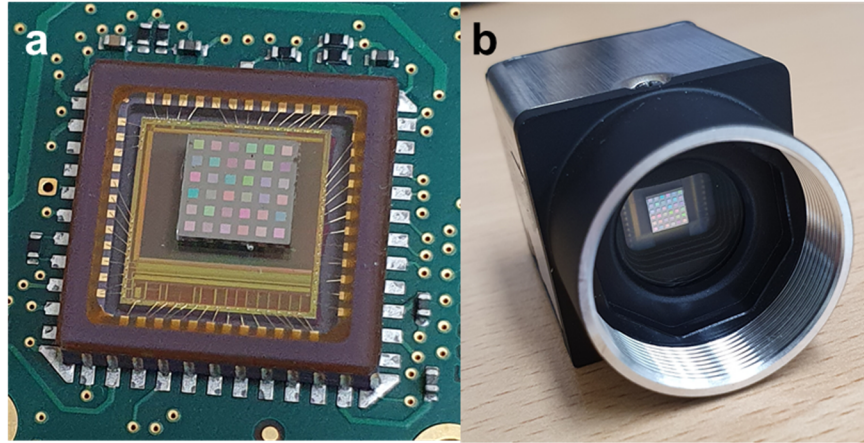


Figure 3.4 MTF filter array-based computational spectrometer. (a) Photograph of the MTF filter array, which is directly attached to the CMOS image sensor. (b) Photograph of the CMOS image camera with the fabricated MTF filter array.

The MTF filter array-based computational spectrometer consists of MTF filters and the CMOS image camera. By attaching the fabricated MTF filter array to the front of a CMOS monochrome camera, the MTF filter array-based spectrometer was built as shown in Fig. 3.4.

3.2.2. Optical Setups for MTF-based Computational Spectrometer

Optical setups for experimental verification of the proposed spectroscopy system are shown in Fig. 3.5. Figure 3.5(a) depicts the optical setup for measuring transmission functions of the MTF filter array. The setup for testing the performance of the proposed system is shown in Fig. 3.5(b). The photographs of the optical setup and the CMOS image camera with the MTF filter array are shown in Fig. 3.5(c). During the optical experiments, we set the incident angle to filter array as normal incidence. Using a linear stage, a rotational stage, and optical mounting posts, we aligned the optical fiber with the CMOS image camera (E0-1312, Edmund Optics) for the normal incidence.

In Fig. 3.5(a), a halogen lamp (KLS-150H-LS-150D, Kwangwoo) was used to provide a continuous light spectrum. It was put into a monochromator (MMAC-200, Mi Optics) to produce a specific narrow wavelength band. Then, a fiber-optic collimator was used to form a beam of parallel light. The beam was fed into the CMOS image camera through the fabricated MTF filter array. With a single exposure, each filter modulated the light in a different pattern. The modulated light was read out by pixels of the CMOS image camera, yielding $M = 36$ distinct output signals y in Eq. (5). Each output signal was taken by summing up the modulated values of the pixels underneath the pertinent filter.

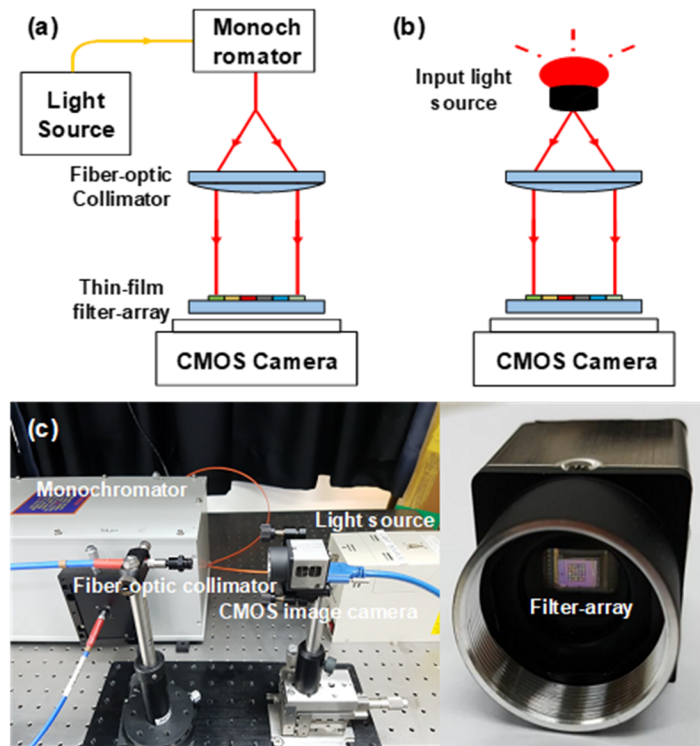


Figure 3.5 (a) Schematic of the optical setup for measuring transmission functions. (b) Schematic of the optical setup for testing the performance of the proposed spectroscopy system. (c) Photographs of the optical setup and the CMOS image camera with the MTF filter array.

3.2.3. Characteristics of MTF-based Computational Spectrometer

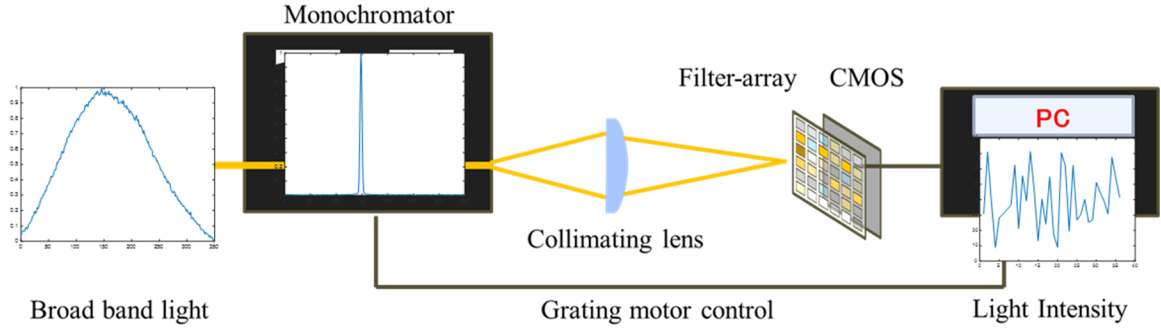


Figure 3.6 The experimental setup for measuring the transmission functions of the fabricated MTF filter array.

Before conducting spectral reconstructions, we first measured the transmission functions of MTF filters. A beam from the halogen light source was fed into a monochromator. From the monochromator, a monochromatic light with an FWHM of 4nm was generated. After passing through a collimator, the collimated monochromatic light was fed into the CMOS monochrome camera. Using the CMOS camera, we measured the light intensities with and without the MTF filter array. Then, the transmission T of i -th filter at wavelength λ_k is calculated by:

$$T_i(\lambda_k) = \frac{IWF_i(\lambda_k) - BI_i(\lambda_k)}{IWO F_i(\lambda_k) - BI_i(\lambda_k)}, \quad (7)$$

where IWF_i , $IWO F_i$, and BI_i are intensity with i -th filter, intensity without i -th filter, and background intensity, respectively. Using the monochromator, we could generate series of monochromatic light at the peak locations from 500 to 849 nm with the step of 1 nm. We captured 350 pairs of monochrome images with and without filters in the wavelength range of 500 to 849 nm. Using Eq. (7), we could obtain transmission functions of 36 MTF filters. Figure 3.7(a) shows the measured two transmission functions of the second MTF filter array. The quantum efficiency of the CMOS image sensor is shown in Fig. 3.7(b).

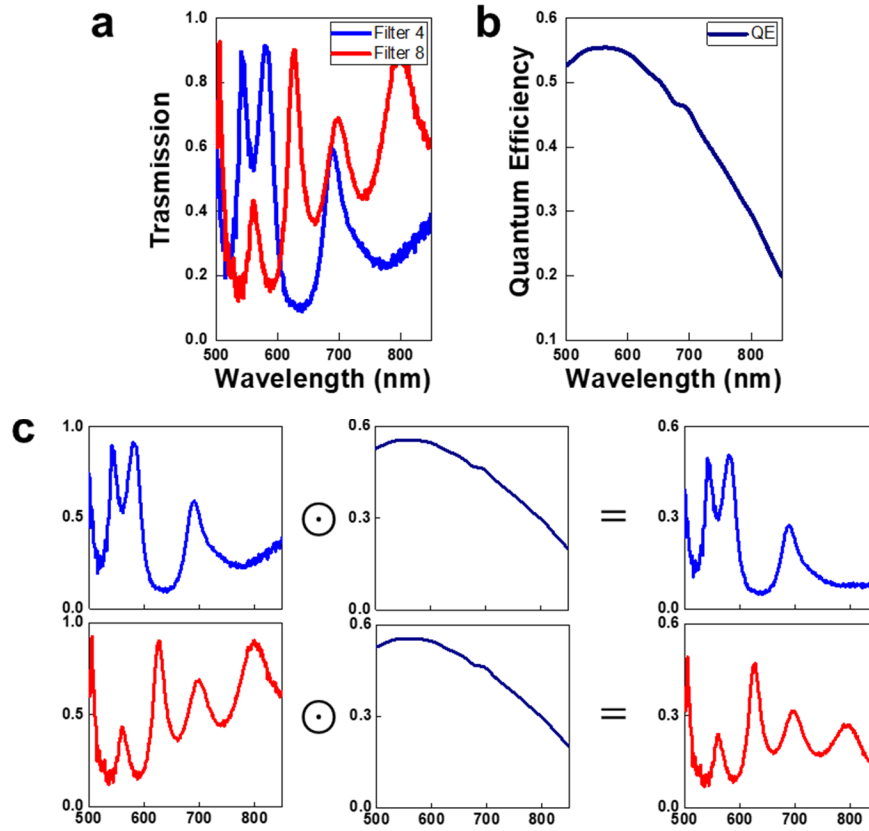


Figure 3.7 (a) Two transmission functions of MTF filters. (b) Spectral response of the CMOS image sensor. (c) Spectral sensitivity of an MTF filter with the CMOS image sensor, which can be calibrated by element-wise multiplication of the transmission function and the spectral response of the CMOS image sensor.

After measuring the transmission functions, the spectral sensitivity of the MTF filter array was calibrated by element-wise multiplication of the transmission functions and the quantum efficiency of the CMOS image sensor as shown in Fig. 3.7(c). Figure 3.8(a) shows the heatmap of the sensing matrix \mathbf{R} of the fabricated second MTF filter array-based computational spectrometer. Each row represents the spectral sensitivity with respect to wavelength. The correlation coefficients for each pair of two rows of the sensing matrix are shown as the upper triangular matrix in Fig. 3.8(b). The average value of the correlation coefficients is 0.231, which can be described as a weak or moderate correlation among sensitivities. With the weakly correlated spectral sensitivities, the incident spectrum was measured as unique intensities, which allow the reconstruction algorithms to work effectively.

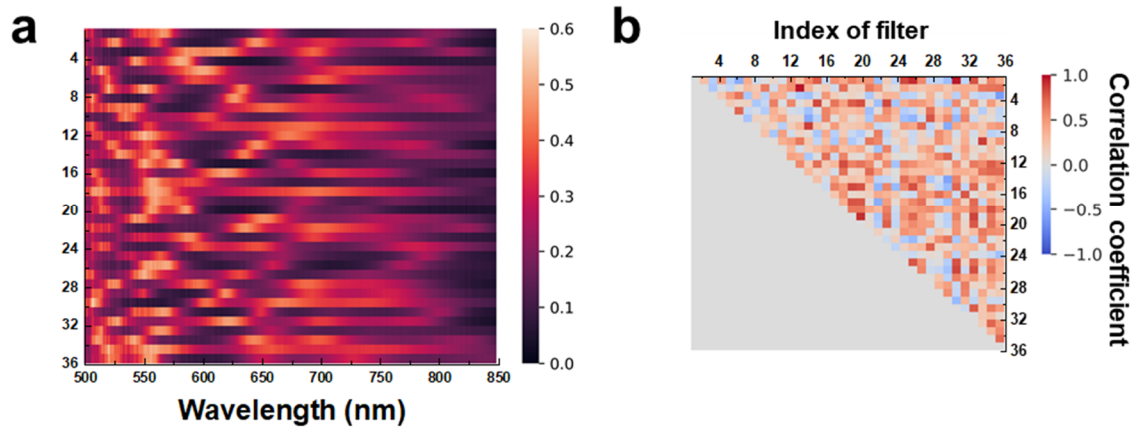


Figure 3. 8 Heatmap of the sensing matrix. Each row represents the spectral sensitivity with respect to wavelength. (b) Upper triangular matrix of correlation coefficients which are pairwise compared among rows of the sensing matrix.

Chapter 4

Spectral Reconstruction

4.1. Spectral Reconstruction using Numerical Optimization

4.1.1. Sparse Recovery

For recovering an unknown spectrum, we use the sparse recovery-based l_1 -norm minimization. \mathbf{x} can be represented as the multiplication of a sparsifying basis $\mathbf{G} \in \mathbb{R}^{N \times N}$ and a sparse signal $\mathbf{s} \in \mathbb{R}^{N \times 1}$, i.e., $\mathbf{x} = \mathbf{G}\mathbf{s}$. Then, Equation (5) becomes $\mathbf{y} = \mathbf{R}\mathbf{G}\mathbf{s}$. The solution of the sparse signal, $\hat{\mathbf{s}}$, can be retrieved by solving the following minimization problem with nonnegativity constraints:

$$\min_{\mathbf{s}} \|\mathbf{y} - \mathbf{R}\mathbf{G}\mathbf{s}\|_2^2 + \gamma \|\mathbf{s}\|_1 \quad \text{subject to } s_k \geq 0 \text{ for } k = 1, 2, \dots, N, \quad (8)$$

where γ is the non-negative regularization parameter and $\|\mathbf{s}\|_p$ is defined as $\left(\sum_{k=1}^N |s_k|^p\right)^{1/p}$. We use the collection of Gaussian distribution functions for the sparsifying basis \mathbf{G} [21]. The linear combination of Gaussian distribution functions represents the line shape of the spectrum. The retrieved spectrum $\hat{\mathbf{x}}$ is $\mathbf{G}\hat{\mathbf{s}}$.

4.1.2. Optical Experiments

We conduct optical experiments using the fabricated two MTF filter arrays. As for the wavelength ranges of interest, we considered 500-1000 nm for the first MTF filter array and 500-850 nm for the second MTF filter array.

Optical experiments of the first MTF filter array. To quantify the performance and explore the two-point resolution of the fabricated MTF filter array, we conducted computational experiments. The two-point resolution is the ability to distinguish the spectral peaks which are closely spaced. For the experiments, we generated mono-peak spectra and two-peak spectra as input spectra using the Gaussian function. A generated input spectrum \mathbf{x} was numerically modulated by multiplying the measured sensing matrix \mathbf{R} as shown in Eq. (1). Then, using the M -modulated signals (measurements) and the sensing matrix \mathbf{R} , a reconstruction algorithm is used to recover the input spectrum. In the experiments, we considered that the input spectrum was a directly sparse signal. The mean-squared error (MSE) between the input spectrum and the reconstructed spectrum $\hat{\mathbf{x}}$ was calculated. The MSE is defined as $\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 / N$.

We firstly tested the spectral reconstruction performance of the fabricated filter-array with changing the full width at half maximum (FWHM) of the generated input signals. We made three noisy environments by adding the additive noise \mathbf{n} to Eq. (5) as $\tilde{\mathbf{y}} = \mathbf{y} + \mathbf{n} = \mathbf{R}\mathbf{x} + \mathbf{n}$ where the signal to noise ratios (SNRs) were 20, 25, 30 dB. The SNR in decibels is defined as $10\log(\|\mathbf{x}\|_2^2 / N\sigma^2)$ where σ is the standard deviation of the noise. The spectral reconstruction performances with respect to the FWHMs are shown in Fig. 4.1(a). For the two-peak spectrum, the distance between two peaks was determined as $[1.5 \cdot FWHM]$, where $[\cdot]$ is the nearest integer function. We averaged all the MSEs of the spectrum over the peak-locations from 500 nm to 999 nm in a given FWHM. As shown in Fig. 4.1(a), the mono-peak spectrum is reconstructed better than two-peak spectrum. As the FWHM increased, the performance of spectral reconstruction is degraded. This is due to the increased sparsity of the spectrum.

Second, we verified the stability of noise along with the SNR conditions for the fabricated MTF filter array. As shown in Fig. 4.1(b), the reconstruction performance on the mono-peak spectrum is better than that of the two-peak spectrum. In addition, when the FWHM is 1 nm, the reconstruction performance is better than the FWHM with 2 nm. Despite the additive noise, the results show that the fabricated filter array is robust to noisy environments.

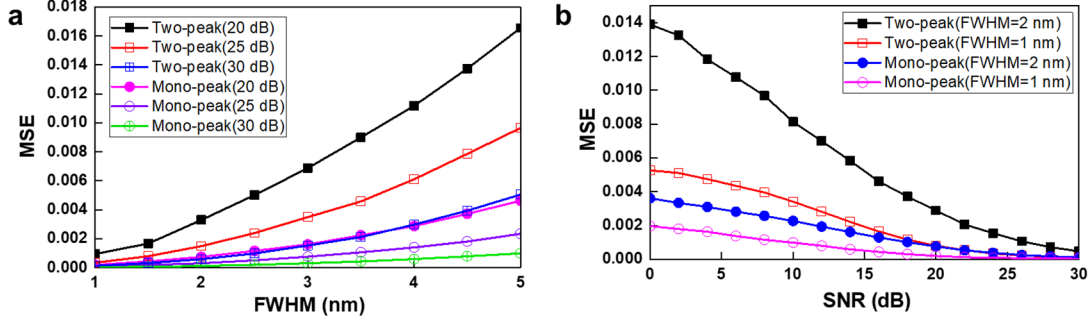


Figure 4.1 (a) Computational reconstruction performance of the fabricated thin-film filter-array with respect to the FWHM. (b) Computational spectral reconstruction performance of the fabricated thin-film filter-array with respect to the SNR.

As depicted in Fig. 4.1, the reconstruction performance of the fabricated filter array depends on the FWHM and the SNR. For the two-point resolution, the MSE has the smallest value when the FWHM of the two-peak spectrum is 1 nm. The overall MSEs are small enough to use the fabricated filter array to conduct the optical experiments.

Optical experiments were then conducted to evaluate the performance of the proposed system. Narrow-band monochromatic lights and LEDs were used as input light sources. To generate narrow-band light, a supercontinuum white light source (SuperK COMPACT, NKT Photonics) was placed in the monochromator, making a narrow band of light with a full width at half maximum (FWHM) of approximately 1 nm. These light sources were fed into the CMOS image camera through the filter array, simultaneously capturing the M differently modulated signals. The M -modulated signals and the measured sensing matrix \mathbf{R} were then used to solve Eq. (8). We used a Gaussian kernel matrix as the sparsifying basis \mathbf{G} . The spectral waveform can be represented as a linear combination of Gaussian kernels, and a Gaussian kernel can be easily generated with two parameters, namely the peak location and the FWHM value [21], [23]. The ll_ls_noneg algorithm [38] was used as a reconstruction algorithm to solve Eq. (8) with non-negativity constraints.

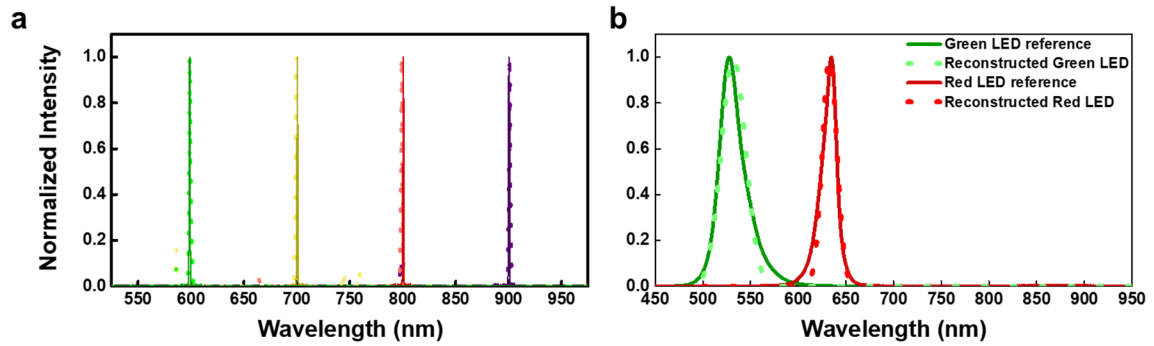


Figure 4.2 Spectral reconstructions of several different input light sources. (a) Spectral reconstructions of monochromatic lights (dots) compared with reference spectra (solid lines): 600 nm (green), 700 nm (yellow), 800 nm (red), and 900 nm (purple). (b) Spectral reconstructions of LEDs (dots) compared with reference spectra (solid lines): green LED (527 nm), and red LED (635 nm).

Figure 4.2 shows the reconstruction results for monochromatic lights and LEDs. For comparison, the reference spectrum and the reconstructed spectrum were normalized to the range between zero and one.

The optical experimental results for monochromatic lights are shown in Fig. 4.2(a). In our optical experiment, we use four different monochromatic spectra, with spectral peaks located at 600, 700, 800, and 900 nm, respectively. The reference spectra are measured using an optical spectrum analyzer (AQ-6315B, Ando) which indicate actual spectral peak locations at 598.7, 700.4, 800.5, and 900.4 nm, respectively. Using the fabricated filter-array computational spectroscopy with the reconstruction algorithm, the spectral peak locations are reconstructed at 599, 699, 799, and 901 nm, respectively. The mean FWHM of the reference spectra is approximately 1 nm, and the mean FWHM for the reconstructed spectra is approximately 1.4 nm.

Figure 4.2(b) shows the spectral reconstructions of green (527 nm) and red (635 nm) LEDs. For the reference spectra, we measure the LEDs using a grating spectrometer (QE65000, Ocean Optics). The spectral peak locations for the reference LEDs are 527.6 nm (green LED) and 634.9 nm (red LED), and the reconstructed spectral peak locations are 531 nm (green LED) and 633 nm (red LED). The peak signal-to-noise ratios are 28.3 dB (green LED) and 31.7 dB (red LED).

Discussing Fig. 4.2, the spectra of reconstructed monochromatic lights show several negligible spikes. This is probably due to background noise in the optical experiments. But overall, the reconstruction results of the first MTF computational spectrometer for monochromatic lights and LEDs are similar to those of the grating spectrometer. Furthermore, the number of modulated signals is significantly small ($M = 36$) that the measurement to wavelength sample ratio is 36:500 (ratio between M and N).

To further explore the performance of the first MTF computational spectrometer, we conducted the computational experiment on the fabricated filter array using a continuous light source, halogen lamp. For the experiment, we used the measured sensing matrix \mathbf{R} . The conventionally measured spectrum of the halogen lamp was used as the input spectrum \mathbf{x} . The modulated signals were generated by numerically multiplying the sensing matrix and the input spectrum. By solving Eq. (8), we reconstructed the continuous spectrum of light. In Fig. 4.3, we present computational spectral reconstruction of the halogen lamp. The peak signal-to-noise ratio is 43.8 dB. Due to the limitations of our optical components to reject the spectrum of the halogen lamp except for the wavelength range from 500 nm to 1000 nm, we could not perform the optical experiment on the continuous source. However, the computational reconstruction result of the halogen lamp indicates that the fabricated filter-array can be utilized for recovering the various kinds of spectra in the given wavelength range without limitations of the optical components.

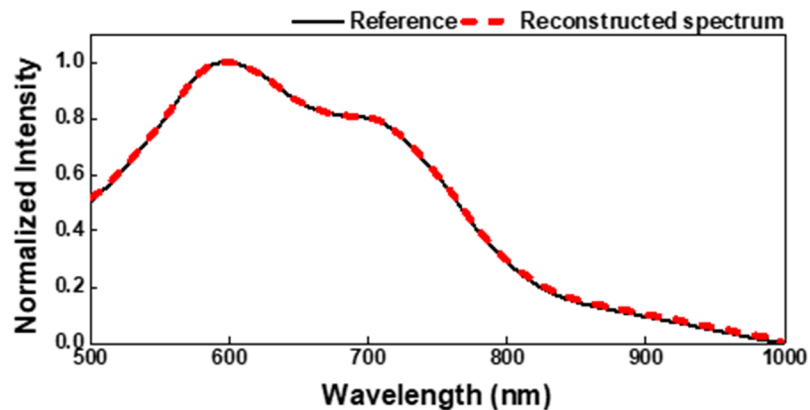


Figure 4.3 Computational spectral reconstruction of a halogen lamp (red dash line) compared with the reference spectrum (black solid line) measured by a conventional spectrometer.

Optical experiments of the second MTF filter array. Before conducting optical experiments, we conducted simulations of dual-peak spectra reconstructions to analyze the dual-peak resolution of the fabricated MTF filter array spectrometer. Using Gaussian distribution functions, we generated a dual-peak spectrum \mathbf{x} as shown in Fig. 4.4(a). The root mean squared error (RMSE), which is defined as $\sqrt{\|\mathbf{x}_{refer} - \mathbf{x}_{recon}\|_2^2 / N}$, was used to evaluate the performances.

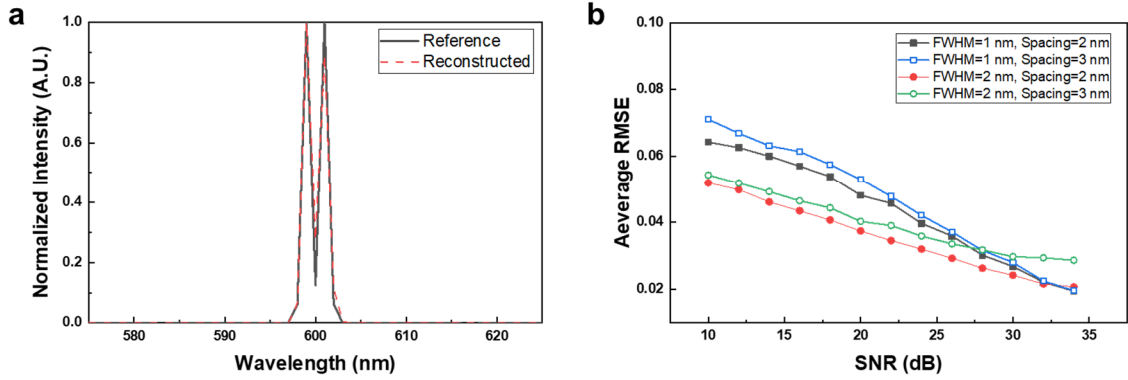


Figure 4.4 Simulation result of two-peaks spectra reconstructions using MTF filter array. (a) An example of two peaks spectra with an FWHM of 1 nm with 2 nm apart. (b) Reconstruction performances of two peaks spectra with respect to signal-to-noise ratios.

The result of dual-peak spectra reconstructions with respect to SNRs is shown in Fig. 4.4(b). We considered four kinds of dual-peak spectra. The FWHMs of a peak were 1 and 2 nm, respectively and the gaps between peaks were 2 and 3 nm, respectively. For each kind of dual-peak, spectra were created by changing the location of dual-peak in the wavelength range of 500 to 849 nm. The reconstructions were performed on all these spectra, and RMSEs were calculated. We averaged the RMSEs and regarded the average RMSE value as the performance of the fabricated MTF filter array to reconstruct dual-peak in noisy environments. As shown in Fig. 4.4(b), the average RMSEs of dual-peak with the FWHM of 1 nm and the gap of 2 nm were 0.0268 for 30 dB, 0.0481 for 20 dB, and 0.0643 for 10 dB, respectively. Similar performances were obtained from the other three kinds of dual-peak spectra. We could find that the fabricated MTF filter array performs well to reconstruct dual-peak spectra in noisy environments.

After analyzing the dual-peak resolution of the MTF filter array in simulations, we tested monochromatic lights by varying the peak wavelength. The CMOS monochrome camera was used for measuring the intensities of test lights. The pixel size of the CMOS image sensor is $5.2 \times 5.2 \mu\text{m}$. Underneath each filter, there are approximately 60×60 pixels. However, considering a case where the layer's location mismatch may occur during the fabrication process of the MTF filter, we excluded the boundary pixels. The averaged intensities from 30×30 pixels at the center of each filter were used for the spectral reconstruction experiments. Using a grating-based spectrometer (Black-Comet, StellarNet), monochromatic lights were measured for use as a reference.

Figure 4.5 shows the reconstruction results for the monochromatic light. For ease of comparison, reference spectra and reconstructed spectra are normalized. Black solid and blue circles in Fig. 4.5 represent reference spectra and reconstructed spectra, respectively. Reference spectra have peak wavelengths at 510, 600, 650, 700, 750, and 840 nm with FWHMs of 4 nm. As depicted in the inset enlarged graph, the reconstructed spectra using the MTF filter array spectrometer matched the reference spectra. More specifically, differences of peak wavelengths between reference and reconstructed spectra were within 2 nm. The RMSEs were 0.023, 0.023, 0.021, 0.035, 0.035, and 0.061, for wavelengths 510, 600, 650, 700, 750, and 840 nm, respectively, as shown in Fig. 4.5(a)-(f). Table 1 presents the evaluation of monochromatic light reconstructions using Gaussian fittings. Over monochromatic lights, peak shifts and FWHMs were within 2 nm and 5.5 nm, respectively. Spectral reconstruction performance seems to degrade in the long-wavelength range due to the low spectral response of the CMOS image sensor and the monotonous spectral features of MTF filters.

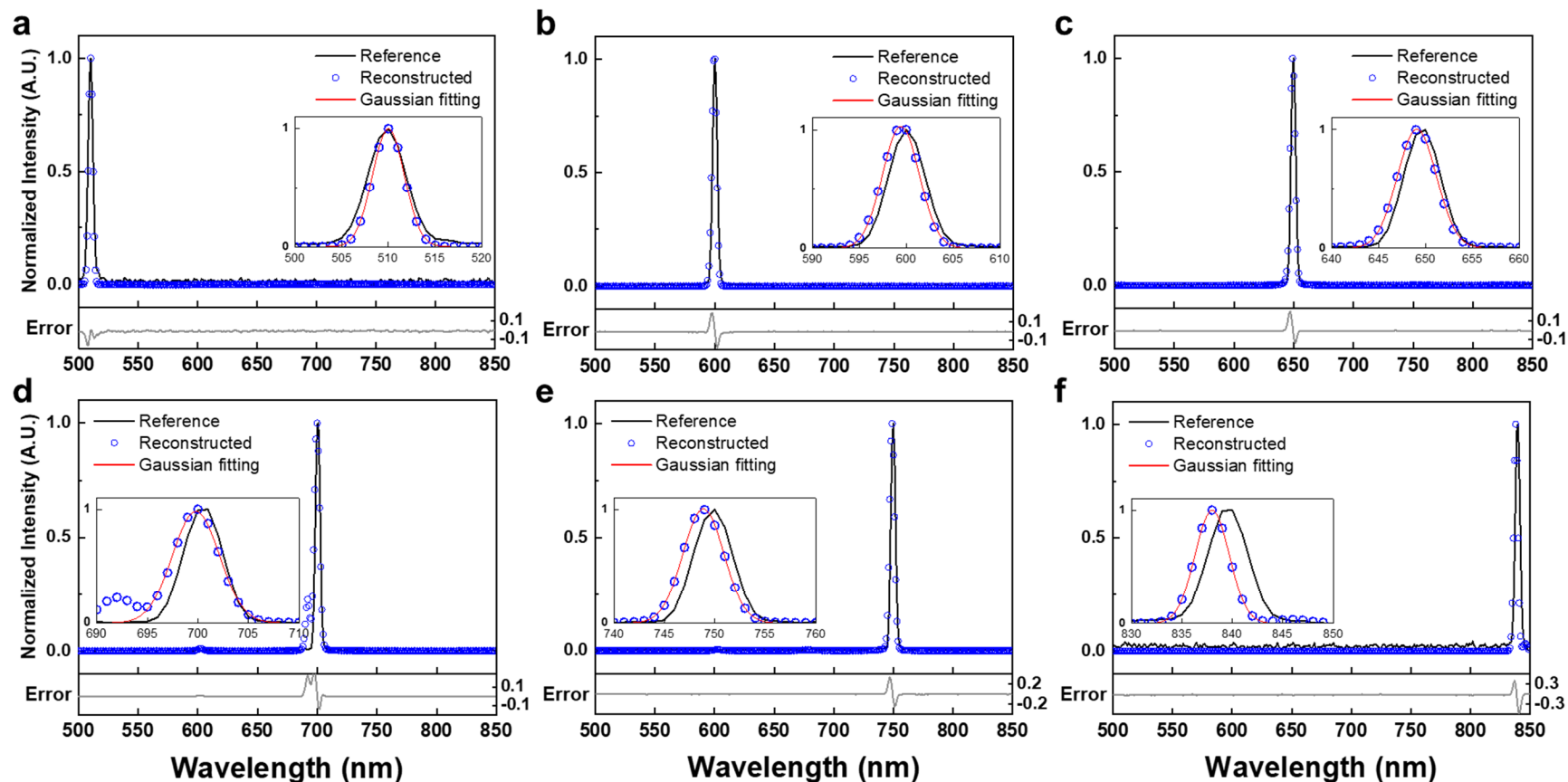


Figure 4.5 Spectral reconstruction of monochromatic light sources. Monochromatic light with an FWHM of 4 nm at peak wavelengths of (a) 510 nm, (b) 600 nm, (c) 650 nm, (d) 700 nm, (e) 750 nm, and (f) 840 nm. Black solid lines represent reference spectra which are measured by the grating-based spectrometer. Blue circles represent reconstructed spectra using the MTF filter array spectrometer. Red solid lines represent the results of Gaussian fitting. Light gray solid lines represent error between reconstructed and reference spectra.

Table 4.1 Evaluation of monochromatic lights reconstructions using the Gaussian curve fitting

Monochromatic light	510 nm	600 nm	650 nm	700 nm	750 nm	840 nm
Peak Center (nm)	509.995	599.438	649.092	699.808	748.878	838.001
Peak Shift (nm)	0.005	0.562	0.908	0.192	1.122	1.999
FWHM (nm)	4.008	4.597	4.887	5.486	4.79	4.003

We further explored the performance of the MTF filter array spectrometer using broadband light sources, such as LEDs and a halogen light source. Figure 4.6 shows the spectral reconstruction results. Black solid lines represent reference spectra, which are measured using the grating-based spectrometer. Colored circles represent reconstructed spectra using the MTF filter array spectrometer. Three single-color visible LEDs and one single-color infrared LED were used for spectral reconstruction experiments, as shown in Fig. 4.6(a)-(d). A green LED (LED 525E, Thorlabs) with an FWHM of 32 nm was reconstructed with an RMSE of 0.021. An orange LED (LED 600L, Thorlabs) with an FWHM of 12 nm was reconstructed with an RMSE of 0.034. A red LED (LED 680L, Thorlabs) with an FWHM of 16 nm was reconstructed with an RMSE of 0.035. An infrared LED (LED 780E, Thorlabs) with an FWHM of 25 nm was reconstructed with an RMSE of 0.044. Similar to experimental results of monochromatic light, the reconstruction performance is relatively poor for spectrum in the long-wavelength range. In addition, we conducted the spectral reconstruction for combined LEDs (an orange LED and a red LED), as shown in Fig. 4.6(e). A beam splitter is used to measure the light of the combined LED. The combined LED was reconstructed with an RMSE of 0.044. Finally, the spectral reconstruction of the halogen light source was conducted. The halogen light source, with an FWHM of 180 nm, was reconstructed with an RMSE of 0.034, as shown in Fig. 4.6(f). As evidenced by low RMSE values, reconstructed spectra agree well with reference spectra measured by the grating-based spectrometer.

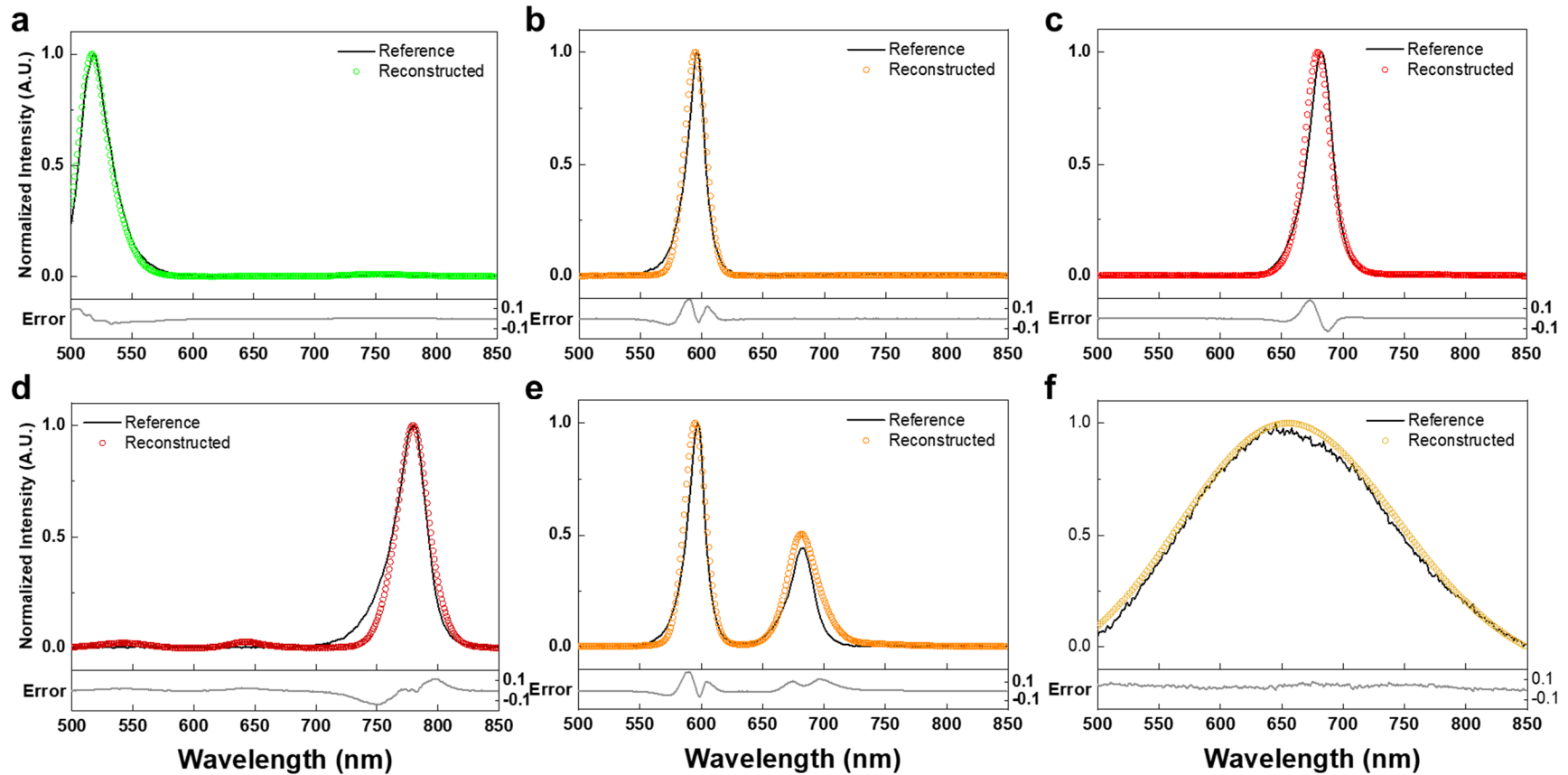


Figure 4.6 Spectral reconstructions of LEDs and a halogen light source. (a) a green LED, (b) an orange LED, (c) a red LED, (d) an infrared LED, (e) Combined two LEDs (orange and red), and (f) a halogen light source. Black solid lines represent reference spectra which are measured by the grating-based spectrometer. Colored circles represent reconstructed spectra using the MTF filter array spectrometer. Light gray solid lines represent error between reconstructed and reference spectra.

4.1.3. Pinhole Imaging

We demonstrated spectral imaging using the MTF filter array. As shown in Fig. 4.7(a), the pinhole imaging system was constructed by combining a pinhole (Edmund optics), whose aperture diameter is 150 μm , with the monochrome CMOS image camera. The MTF filter array was placed in front of the pinhole. A single filter was adjusted to the pinhole to allow an incident image to pass through the filter and pinhole, and the filtered image was measured by the CMOS image sensor. By changing filters using a linear translation stage (Newport), 36 filtered images are obtained. Bi-Color 8 \times 8 LED matrix (Adafruit) was used to generate a target. We made a small display by connecting the LED matrix to an Arduino Uno (Arduino) and by controlling the color of the 64 blocks. The number “8” was represented by the LED cube. The upper blocks consist of green LEDs, and the lower blocks consist of red LEDs. Figure 4.7(b) shows a stack of the filtered 36 sub-images. A 1280 \times 1024 size image was reduced to a sub-image size of 350 \times 300 by discarding unnecessary pixels. Thus, a data cube with 350 \times 300 \times 36 in size was obtained. Spectral reconstruction was performed for each pixel, and the data cube was restored with a size of 350 \times 300 \times 350. It took \sim 1.8 h to reconstruct the data cube. As shown in Fig. 4.7(c), the reference spectra measured by the grating-based spectrometer are shown in solid black lines. As denoted pixels in Fig. 4.7(b), the reconstructed spectra of a pixel in the green LED block and a pixel in the red LED block are represented in Fig. 4.7(c) as green circles and red circles, respectively. The RMSE was calculated after normalizing reference spectra and reconstructed spectra. The green LED with an FWHM of 15 nm was reconstructed with an RMSE of 0.0315. The red LED with an FWHM of 20 nm was reconstructed with an RMSE of 0.0370. Figure 4.7(d) shows the monochrome image of reference and reconstructed monochrome images at 571, 600, and 638 nm. The pinhole imaging system also measured the monochrome image of reference without the MTF filter array. Since the spectral component of the red LED does not exist at 571 nm, only the upper blocks of the number “8” are shown in the reconstructed monochrome image at 571 nm. On the other hand, only the lower blocks of the number “8” are shown in the reconstructed monochrome image at 638 nm, where the spectral component of the green LED does not exist. Finally, nothing is displayed in the reconstructed monochrome image at 600 nm, where spectral components of the green and red LED do not exist.

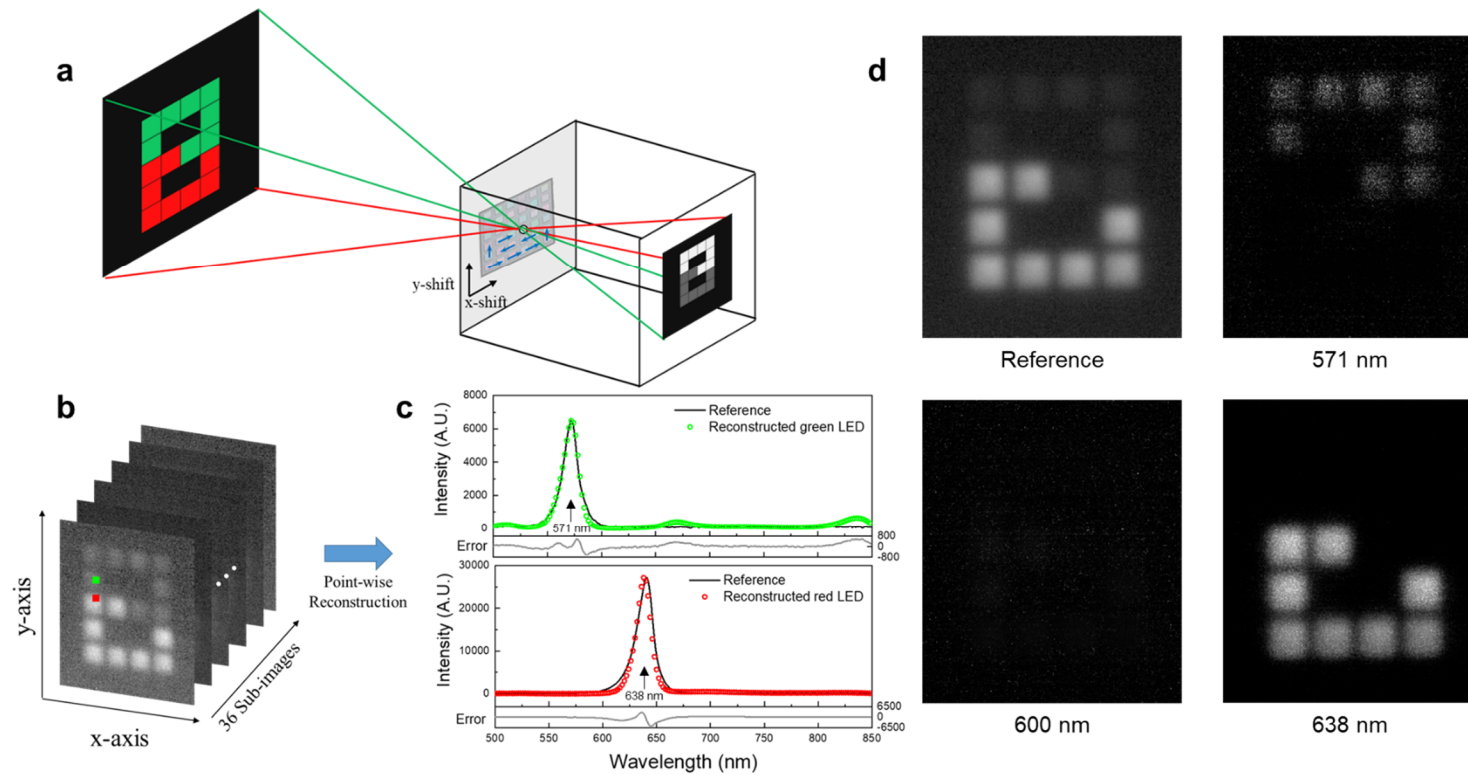


Figure 4.7 Computational pinhole spectral imaging. (a) Schematic of pinhole imaging; The MTF filter array is placed in front of the pinhole camera. A single filter is adjusted to the pinhole and the filtered image is acquired. By changing filters, 36 filtered images are obtained. (b) 36 filtered images of 8×8 LED matrix showing the number “8”; The upper part consists of green LEDs and the lower part consists of red LEDs. (c) Point-wise spectral reconstruction: a pixel of a green LED block and a red LED block which are denoted in (b). Light gray solid lines represent error between reconstructed and reference. (d) Monochrome image of reference and reconstructed monochrome images at 571, 600 and 638 nm, respectively.

As proof-of-principle of the spectral imaging, we implemented the spectral scanning method on the pinhole imaging system. While reconstructed spectra of the pinhole spectral imaging match well with reference spectra, there are improvements to consider. In the spectral scanning method, the data cube acquisition time is long so that spectral smearing may occur in the case of a moving target. The non-scanning method, such as in snapshot spectral imaging systems, can solve these problems by acquiring the data cube in a single exposure. We assume that it is possible to construct a snapshot spectral imaging system combining the MTF filter array and a thin observation module by bound optics (TOMBO) [48]–[50] structure. This spectral imaging system requires a microlens array and a single separator but does not need the MTF filter to be as small as pixel size. Rather, the MTF filter should be made large so that many pixels are underlying the filter. The MTF filter array can be fabricated in scalable using stencil lithography techniques according to the spatial resolution of the spectral imaging system.

4.1.4. Summary

In summary, we fabricated two kinds of MTF filter arrays using lithography techniques and experimentally demonstrated the spectral resolvability of the two MTF filter array-based computational spectrometers. Although the MTF filter size was larger than that of the photonic crystal slabs [29], [33], it can be improved to a smaller size by using advanced lithography techniques and facilities. In addition, by using a higher refractive index material, the number of layers of the MTF filter can be reduced so that manufacturing efficiency can be improved.

Using the random spectral features of MTF filters and numerical optimization techniques, we recover varied spectra from the visible range to the near-infrared range with 1 nm spacing. We tested light sources, including monochromatic lights, LEDs, and a halogen lamp. We showed good recovery performances. The spectral reconstruction performance in the near-infrared range is relatively inferior to the visible range, but it can be further improved by using a CMOS image sensor with a high spectral response in the near-infrared region. Also, computational spectral imaging with the MTF filter array was demonstrated using the spectral scanning method. The reconstructed data cube was found to match well with spatial and spectral references. However, to use the spectral imaging system in mobile

applications, a shorter data cube acquisition time is required. By utilizing the TOMBO structure with the MTF filter array, it is possible to construct a snapshot spectral imaging system that has a short acquisition time.

Finally, the mass production of the MTF filter arrays is an important step towards the industrialization and practical uses of computational spectrometers. This study will be helpful for computational spectroscopy to be realized in various mobile applications, such as on-site detection, UAV-based remote sensing, and skincare.

4.2. Spectral Reconstruction using Deep Learning

4.2.1. Deep Learning Architecture for Spectral Reconstructions

Computational spectrometers mostly adopted numerical optimization approaches based on constraints that the light sources are sparse signals, or can be sparsely represented by a certain sparsifying basis. However, there is a limitation to sparsely representing all spectra with a fixed sparsifying basis. In addition, these approaches work well for precisely measured signals and handcraft parameters predetermined through prior information, such as sparsifying basis, line shapes, and full width at half maximums of spectra. Thus, the recovery performance of a spectrum could be biased depending on the variation in noise level and predetermined parameters. These limitations make it difficult to use computational spectrometers for practical and mobile applications.

In this section, we propose deep learning (DL)-based computational spectrometer. As a configuration of the computational spectrometer, we employed the second MTF filter array of 36 filters and a CMOS image camera. The MTF filter array was fabricated through a wafer-level stencil lithography process, which can be scalable, reproducible, and mass-produced. By directly attaching the MTF filter array to the CMOS image camera, the computational spectrometer was built. The incident light was modulated by the MTF filter array, where each filter had a unique transmission function. The filtered light through the MTF filter array was measured by the CMOS image camera with a single exposure. From the captured image, we extracted 36 compressively sampled light intensities. The

intensities were fed into a DL architecture and reconstructed as a spectrum of 350 elements. We collected 2,873 continuous spectra for training/validation/testing the DL architecture. After training, we tested the DL architecture performances using unused data in the training/validation process. The average root-mean-squared errors (RMSE) was 0.0237. The results of performance evaluations demonstrate that the proposed DL architecture performed well to continuous spectra. The proposed spectrometer is compact with a single-shot structure and can be mass-produced. Besides, by applying the DL technique, it can have a high resolution, wide working range, fast reconstruction. Therefore, the proposed spectrometer can become a new form factor for on-site detection, such as food inspection, counterfeit document detection, and self-diagnosis.

Figure 4.8(a) depicts a schematic of the experimental setup with the proposed spectrometer. A collimating beam from a halogen lamp is divided into two beams after passing through a color filter and a long pass filter. A spectrum of one split beam was measured using a commercial spectrometer and it was used as the ground truth (GT). The other beam was fed into the MTF filter array and modulated by the transmission functions of filters. The modulated intensities of the beam were captured by a CMOS image camera as a monochrome image with a single exposure. By connecting the spectrometer and CMOS image camera to a laptop using universal serial bus cables, we simultaneously collected the monochrome image and GT spectrum. The captured monochrome image had a size of 1280×1024 pixels, and the ground truth spectrum is a signal of 350 elements measured at the wavelength range of 500-850 nm with 1 nm spacing.

As shown in Fig. 4.8(b), we extract 36 intensities from the monochrome image. These intensities were fed into the DL architecture to be a reconstructed spectrum. The proposed DL architecture, residual convolutional neural network (RCNN), comprises a learnable sensing matrix and sets of convolution and dense blocks, and a residual connection. We trained the RCNN using the training dataset and test the resolving performance using the test dataset. Figure 4.8(c) shows examples of reconstructed test spectra using the trained RCNN. The RMSE between GT spectra (black dashed lines) and reconstructed spectra (solid green lines) was used as the performance evaluation metric. The reconstructed spectra agreed well with the GT spectra, as shown by the RMSE values written in the upper left of each plot (Fig. 4.8(c)).

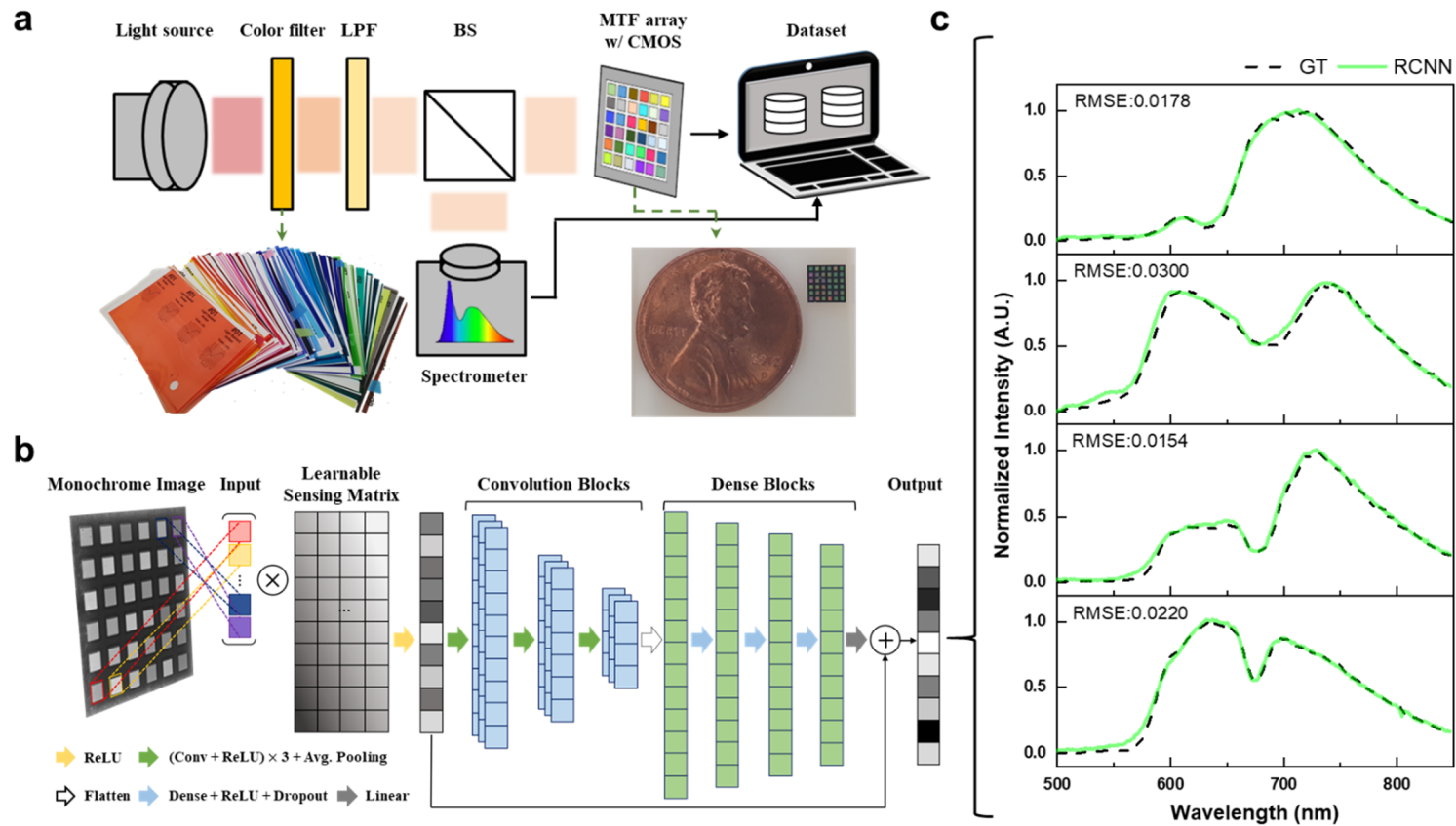


Figure 4.8 DL-based single-shot computational spectrometer. (a) Schematic of the experimental setup. (b) DL architecture, RCNN, comprises convolution blocks, dense blocks, and a residual connection with a learnable sensing matrix. (c) Examples of reconstructed test spectra using a trained RCNN. The black dashed line represents ground truth spectra measured using a commercial spectrometer. The solid green line represents reconstructed spectra using the proposed spectrometer.

Figure 4.8(b) depicts a schematic of RCNN. The 36 measured intensities were extended to a size of 350 by multiplying the transpose of the learnable sensing matrix. This extension allowed deep stacking of convolution blocks, which could be helpful in feature extraction. We used the sensing matrix as the initial value of the learnable sensing matrix. After going through a rectified linear unit (ReLU) activation function, the extended signal was fed into the sets of convolution and dense blocks. A convolution block comprised three pairs of convolutional layer and ReLU, and one average pooling layer. A series of three convolution blocks were stacked to perform feature extraction at various levels from the extended signal. The output of convolution blocks was flattened and fed into a set of dense blocks. A dense block comprised a dense layer, ReLU, and a dropout. A series of three dense blocks and a dense layer with a linear activation function were stacked to perform the spectral reconstruction. The output of dense blocks and the extended signal were summed by a residual connection to become a reconstructed spectrum.

RCNN was trained to minimize a mean squared error between the reconstructed and GT spectra. By leveraging the residual connection, the stack of blocks learned the *residue* between extended signals and GT spectra. It was reported that learning *residue* is more effective than directly learning target spectra [41]. In addition, the residual connection prevents the gradient vanishing problem, which could stop the updating of DL architectures in the training process [51].

For the hyperparameter selection, we monitored the performance of RCNN for every epoch during the training process using a validation dataset. As such, we could select the number of epochs before overfitting. The selected number of epochs was 500. The batch size and learning rates was 8 and 0.0005. In addition, we used regularization techniques, such as the l2-layer weight regularizer with 0.5, and dropout with 0.5. The number of filters and kernel size of all convolution layers was 64 and 3, respectively. The number of units in the four dense layers was 2,048, 1,024, 512, and 350 in that order (Fig. 4.8(b)).

4.2.2. Experimental Dataset

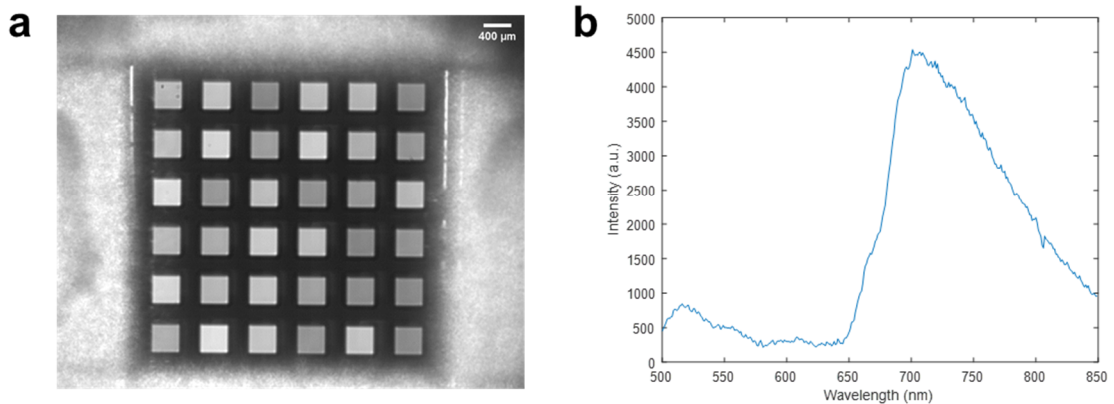


Figure 4.9 Example of INFONET Colors dataset. (a) monochrome image illuminated by a light source. (b) corresponding spectrum of a light source.

Table 4.2 INFONET Colors dataset.

Dataset	Training / Validation / Test	Description
INFONET Colors	2296 / 289 / 288	Continuous spectra generated by modulating a halogen source using color filters

To train RCNN, we collected datasets comprising 2,873 pairs of monochrome images and corresponding GT spectra as shown in Fig. 4.9. By changing the combination of color filters (Color filter booklet, Edmund optics) in the experiment setup in Fig. 4.8(a), the spectrum of the halogen lamp was transformed into the spectrum of various waveforms. The CMOS image camera and halogen lamp were calibrated to capture the intensity of the halogen lamp in the range of 0 to 255, and the CMOS image sensor’s autocontrast function was turned off. The spectra were measured with a fixed integration time. As shown in Table 4.3, we randomly divided the dataset into training, validation, and testing datasets with the number of 2296, 289, and 288, respectively. Before training RCNN we conducted preprocessing of the dataset. The measured intensities from a monochrome image were divided by the maximum value of intensities, and the corresponding GT spectrum was min-max normalized. Therefore, we trained the RCNN to reconstruct the unknown spectra

in the form of the normalized intensity. We used the Adam optimizer[52] for training RCNN. The training process was completed within ~ 20 min. The trained RCNN was tested on the test dataset, and the test results came out within ~ 2 s. The training and testing were conducted on an Intel Core i7-5820K CPU computer with an NVIDIA GeForce RTX 2060 graphics processing unit.

4.2.3. Results

Figure 4.10 shows the reconstruction results of the proposed DL-based computational spectrometer. The RMSE distribution of 288 test spectra is shown as a histogram in Fig. 4.10(a). The average RMSE of test spectra was 0.0237. Three examples of the best and worst spectral reconstructions, denoted by a green box and blue boxes in the histogram, are shown in Fig. 4.10(b-c), respectively. The black dashed lines represent GT spectra and the solid colored lines represent reconstructed spectra. Based on the RMSE distribution in the histogram, RCNN recovered the test spectra well. Considering the worst examples, it seems that the reconstruction performances deteriorated due to poor signal quality and low intensity. In addition, the normalization of GT spectra of the low light intensity amplifies the noise in spectra, which deteriorated the reconstruction performances of RCNN.

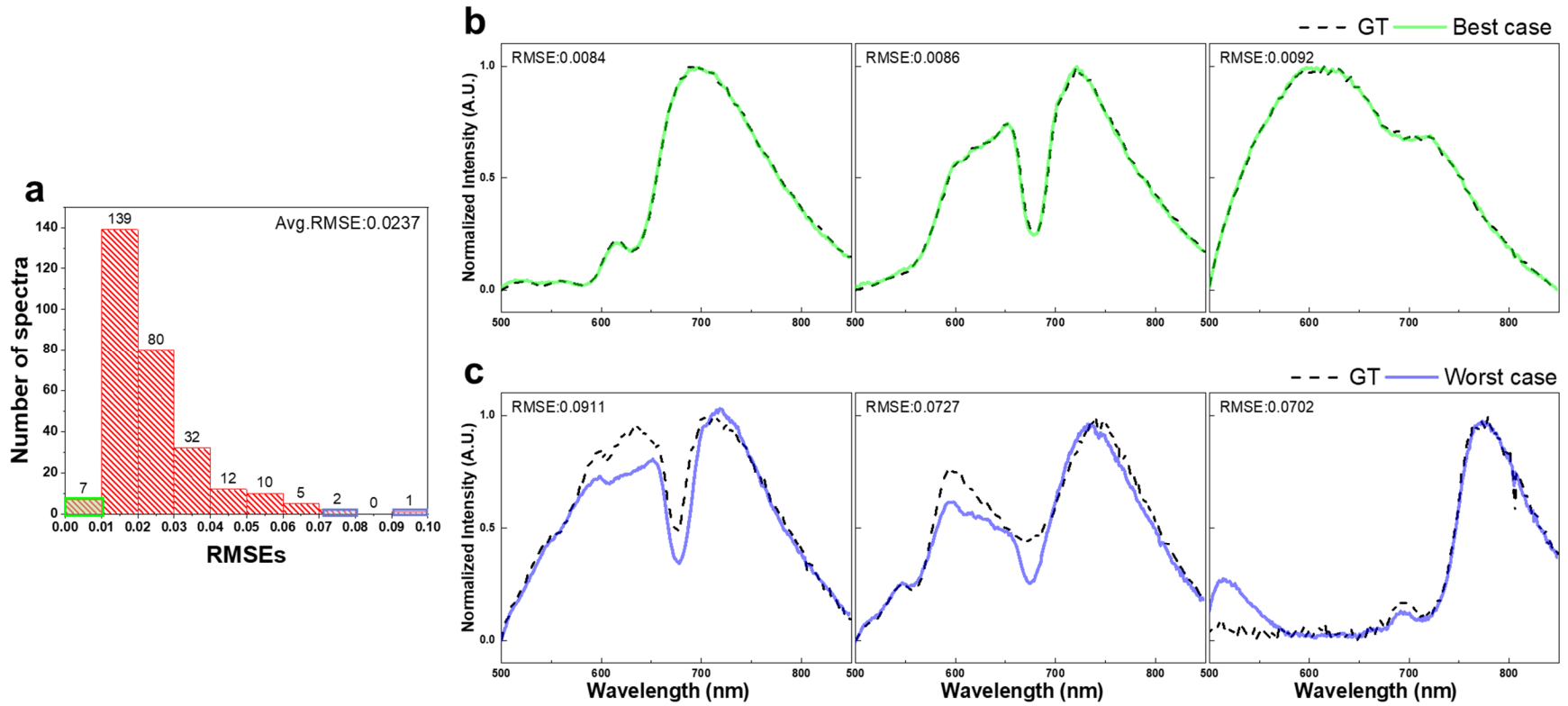


Figure 4.10 Result of continuous spectra reconstructions. (a) Histogram of the RMSE distribution for the test dataset; average RMSE is 0.0237. (b) Three examples of the best spectral reconstructions, which are denoted as a green box in the histogram. (c) Three examples of the worst spectral reconstructions, which are denoted as blue boxes in the histogram. The black dashed lines represent GT spectra. The solid colored lines represent reconstructed spectra.

Instead of training the RCNN using the normalized dataset, we additionally trained the RCNN using a scaled dataset. The datasets of measured intensities and GT spectra were divided by the maximum value from each dataset, respectively. e.g. 255 value from the monochrome image. Therefore, the datasets were transformed to fit within the scale of [1, 0] and had information of relative intensities, which were not contained in the normalized dataset.

Figure 4.11 shows the results of spectral reconstructions using the normalized and scaled datasets. The normalized and scaled test intensities extracted from monochrome images were compared with references (Fig. 4.11(a)). The references are simulated intensities calculated by multiplying GT spectra and the sensing matrix. For comparison ease with the normalized intensities, a reference was normalized. The solid red, black, and gray lines represent references, normalized intensities, scaled intensities, respectively. Although there were slight differences between the reference and normalized intensities, we could figure out that the waveform was matched with a good agreement. Differences may occur due to noise that can be measured during optical experiments and light intensity outside the wavelength range of interest. Figure 4.11(b) shows the reconstructed spectra using RCNN trained with the normalized dataset. The reconstructed spectrum of each row in Fig. 4.11(b) was obtained using the normalized test intensities of each row in Fig 4.11(a). The reconstructed spectra (solid green lines) are matched well with the GT spectra (black dashed lines). In addition, they were reconstructed to normalized spectra, and the waveforms of the spectra were depicted in detail. The reconstructed spectra using RCNN trained with the scaled dataset are shown in Fig. 4.11(c). The reconstructed spectra of each row were obtained using the scaled test intensities of each row in Fig. 4.11(a). The reconstructed spectra (solid blue lines) matched well with the GT spectra (black dashed lines). The relative intensities among the reconstructed spectra can be figured out.

The trained RCNN with the normalized and scaled dataset can be employed differently depending on the applications. For example, the trained RCNN with the normalized dataset can be used for classification-related applications because it can recover spectra in detail, whereas the trained RCNN with the scaled dataset can be used for analytical applications where the relative intensity information is crucial, e.g., hyperspectral imaging.

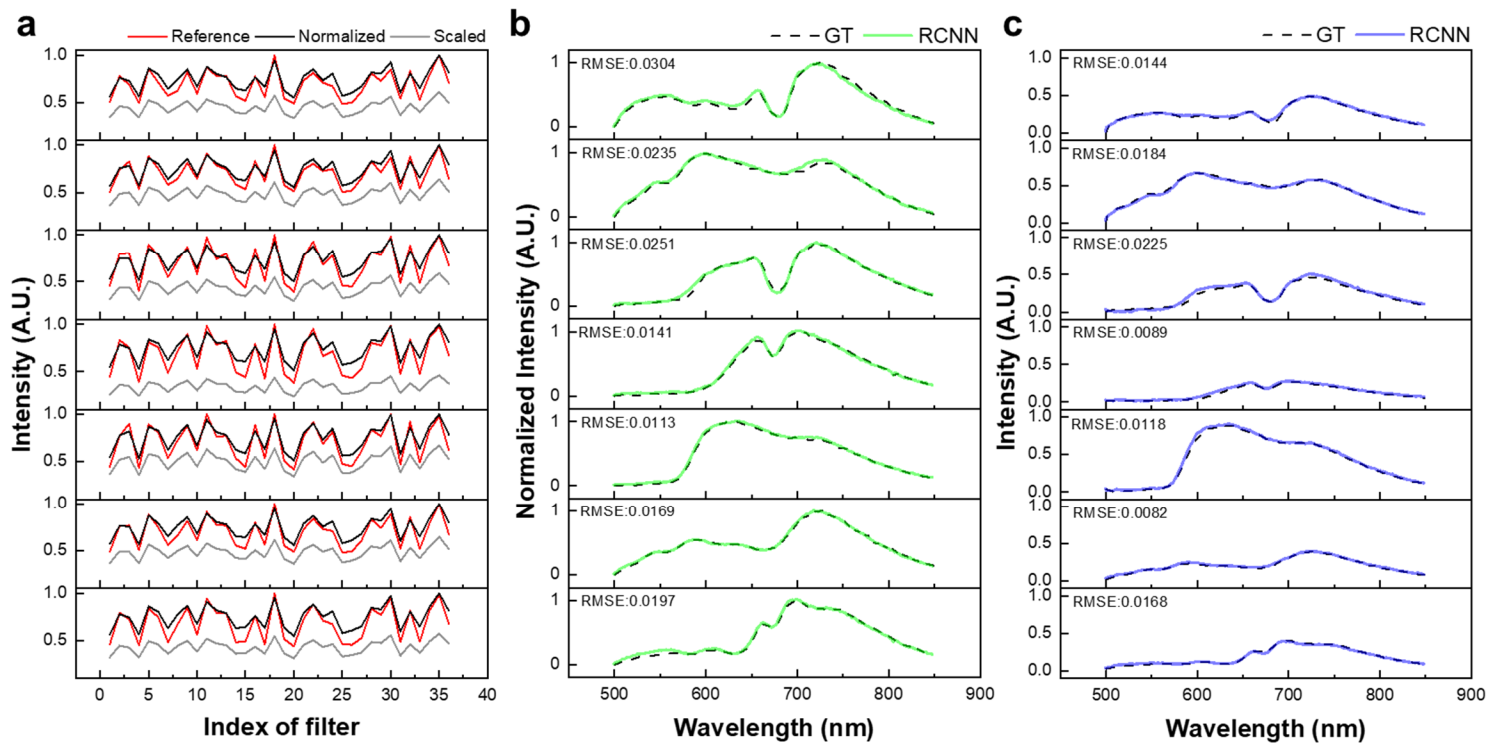


Figure 4.11 Results of spectral reconstructions using the normalized and scaled datasets. (a) Test intensities extracted from monochrome images in real experiments are compared with references. The solid red lines represent normalized references for easy comparison with the normalized intensities (solid black lines). The solid gray lines represent scaled intensities. (b) Results of spectral reconstructions from the trained RCNN (solid green lines) using the normalized dataset compared with GT spectra (black dashed lines). (c) Results of spectral reconstructions from the trained RCNN (solid blue lines) using the scaled dataset compared with GT spectra (black dashed lines). The reconstructed spectra of each row in (b) and (c) were obtained using the normalized and scaled test intensities in each row of (a), respectively.

We conducted additional experiments for reconstructing transmission spectra of drinks using the RCNN trained with the normalized dataset. The samples of drinks were prepared using disposable polystyrene cuvettes capacity of 4 ml (Fig 4.12(a)). The pink, yellow, light yellow, blue, purple drinks are Vitaminwater[53]-Power C, Energy, Powerade[54]-Mega Boltz, Mountain Blast, and Purple Storm, respectively. By replacing the color filters with the drink samples, the monochrome images and transmission spectra of drinks were measured using the experimental setup depicted in Fig. 4.8(a). The reconstructed transmission spectra of drinks in Fig. 4.12(a) are illustrated in Fig. 4.12(b–f). The black dashed lines represent the GT spectra and the solid colored lines represent the reconstructed spectra using RCNN. The solid gray lines represent the reconstructed spectra using sparse recovery. The RMSEs for each drink is written in the upper left corner of each graph. The average RMSEs of drinks was 0.0597 for RCNN, and 0.1648 for sparse recovery. RCNN shows better reconstructions than sparse recovery. The RMSEs of transmission spectra of drinks were lower than the reconstruction of color filter spectra because the transmission spectra through cuvettes with the light path of 10 mm filled with drinks were reconstructed using the RCNN trained using thin color filters but still showed acceptable performances.

Although there is still room for improvements, we demonstrated that the proposed DL-based computational spectrometer is applicable to the reconstruction of transmission spectra of drinks. Since complex optical components or long light paths are not required, the optical system (Fig. 4.8(a)) can be made compact. In addition, the MTF filter array can be mass-produced at low cost through wafer-level stencil lithography processing, so it can be fabricated as an inexpensive and compact sensor for mobile applications, such as on-site detection and simple diagnostic tests.

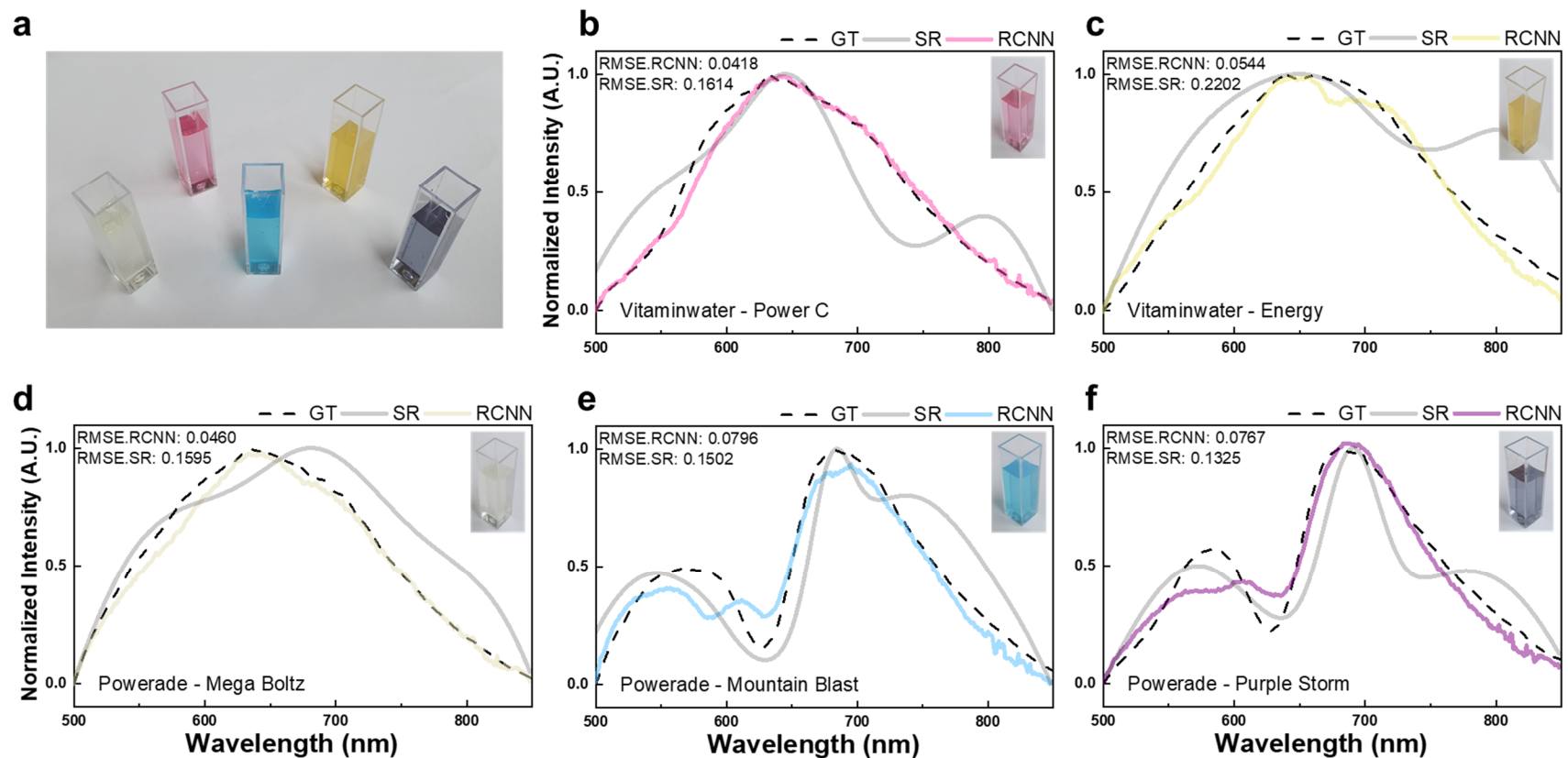


Figure 4.12 Reconstructions of transmission spectra of drinks. (a) Photograph of drink samples: Vitaminwater-Power C (pink), Vitaminwater-Energy (yellow), Powerade-Mega Boltz (light yellow), Powerade-Mountain Blast (blue), and Powerade-Purple Storm (purple). (b–f) Reconstructed transmission spectra of drinks. The black dashed lines represent the GT spectra, and the solid colored lines represent the reconstructed transmission spectra using RCNN. Solid gray lines represent the reconstructed spectra using sparse recovery.

4.2.4. Summary

In this section, the DL-based computational spectrometer using the MTF filter array was proposed. The proposed spectrometer has advantages in two main aspects: hardware and software features. The footprint of the proposed spectrometer was compact by attaching the MTF filter array to the CMOS image camera. This is useful for mobile applications as it measures all filtered intensities in a single exposure. The MTF filter array can be mass-produced at low cost by wafer-level stencil lithography processing. Unlike the conventional bandpass filters that transmit a specific light wavelength range, the MTF filters transmit light in a wide wavelength range, so the filters had a good light efficiency. In addition, the filters can provide spatial information, which can be developed into a hyperspectral imaging system.

We reconstructed the continuous spectra with high resolution in the wide wavelength range (500–850 nm with 1 nm spacing) using the DL architecture RCNN. By adopting DL, the number of filters for covering the wide wavelength range could be reduced. Although there are iterative numerical optimization approaches for computational spectrometers, only a few cases showed the reconstruction for continuous spectra. In addition, iterative approaches work well for precise and fewer noise measurements otherwise. There are cases in which spectral reconstructions work well by selecting handcraft parameters using the prior information of a target spectrum, but it is difficult to find parameters that work for multiple spectra. Moreover, although RCNN requires training datasets, it is trained end-to-end without the need for human intervention. It took ~20 min to train RCNN with 2,296 training spectra. The trained RCNN reconstructed 288 test spectra with an average RMSE of 0.0237 in ~ 2 s.

Although there is a limitation that the spectra are confined to continuous spectra, we could extend the spectra dataset using a monochromator to generate narrow peak spectra with different wavelength locations.

In summary, we developed an MTF filter array spectrometer with RCNN. RCNN was trained to reconstruct the normalized or scaled spectra depending on the preprocessing of datasets. The reconstruction performance was experimentally demonstrated using the

spectra of color filters and drinks and showed good performances. We expect that the hardware of the proposed computational spectrometer can be fabricated to be scalable, and the software can be trained to be flexible in compliance with spectral applications.

4.3. Comparison between Numerical Optimization and Deep Learning

4.3.1. Motivation

Since not all signals can be represented as sparse signals on a fixed basis, therefore prior information on structural features of the spectral dataset is required to generate a best fit sparsifying basis in numerical optimization approaches. Furthermore, a high computational cost is required for numerical optimization approaches. Deep learning [11] has been emerging as a promising alternative framework for reconstructing the original signal from the compressed measurements.

Mousavi et al. [55] was the first study on image recovery from structured measurements using deep learning. Moreover, a deep learning framework for inverse problems has been applied in biomedical imaging for imaging through scattering media [56], magnetic resonance imaging [41], [57], and X-ray computed tomography [58]. Kim et al. [18] reported the first attempt to use deep learning in computational spectroscopy. They trained a convolutional neural network (CNN) to output the reconstructed signal from the network. From here on the network reported by Kim et al. will be referred to as CNN.

Unlike CNN [18] in which learnable layers were simply stacked and trained to directly reconstruct the original spectrum, we make a residual connection [51] between the input and output of CNN and train the network to reconstruct the original spectrum by referring to the input of the network. As a result, the network learns residuals between the input of the network and the original spectrum. It has been reported that it is easier to train a network when using residual connections than to train a plain network that was simply stacked with learnable layers [41], [51]. Lee et al. [41] analyzed the topological structure of magnetic resonance images and the residuals of MR images. They showed that the residuals possessed a simpler topological structure, thus making learning residuals easier

than learning the original MR images. In addition, He et al. [51] demonstrated with empirical results that the residual networks are easy to optimize and they achieved improvements in image recognition tasks. From these works, we hypothesize that adding a residual connection to CNN will improve the spectral reconstruction performance in computational spectroscopy.

We propose a novel residual convolutional neural network (RCNN) for recovering an input spectrum from the compressed measurements in computational spectroscopy. The performance of the proposed RCNN was demonstrated using numerical experiments using synthetic and measured datasets. We compared the reconstruction performance of RCNN with those of existing reconstruction approaches such as reconstruction algorithms with a sparsity constraint, and CNN. Compared with the reconstruction algorithms, RCNN shows better reconstruction performances without the a priori knowledge of either a sparsifying basis or any spectral features of the spectral datasets. Moreover, RCNN shows stable reconstruction performances under noisy environments. Finally, compared to CNN, RCNN renders superior reconstruction performance and RCNN is trained faster than CNN.

4.3.2. Deep Learning Architecture

Typically, except for narrow-band spectra, a spectrum is not a sparse signal, and a fixed sparsifying basis cannot transform all spectra into sparse signals. Clearly, the use of a fixed basis may lead the sparse recovery to struggle, as no fixed basis will transform every signal into a sparse signal. In addition, the sparse recovery is time-consuming and takes a high computational cost.

Our goal is to overcome the limitations of the sparse recovery in computational spectroscopy and recover various kinds of spectra using RCNN. Figure 4.13 shows the schematic of computational spectroscopy system using RCNN. This system consists of two parts: the compressive sampling and dimension extension, and the reconstruction using RCNN. In the compressive sampling and dimension extension, the measurement vector \mathbf{y} is obtained from Eq. (5), which then transforms into $\tilde{\mathbf{x}} \in \mathbb{R}^{N \times 1}$ using a linear transformation. A transform matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ extends the M dimension of \mathbf{y} to N dimension of $\tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}}$ is a representative spectrum corresponding to \mathbf{x} . We used $\tilde{\mathbf{x}}$

as the input for the reconstruction. RCNN learnt a non-linear mapping between $\tilde{\mathbf{x}}$ and \mathbf{x} , and afforded a reconstructed spectrum $\hat{\mathbf{x}} \in \mathbb{R}^{N \times 1}$. The dimension extension by the transform matrix was used to make it easier for RCNN to extract features and reconstruct spectra from the non-linear mapping.

Spectroscopy system

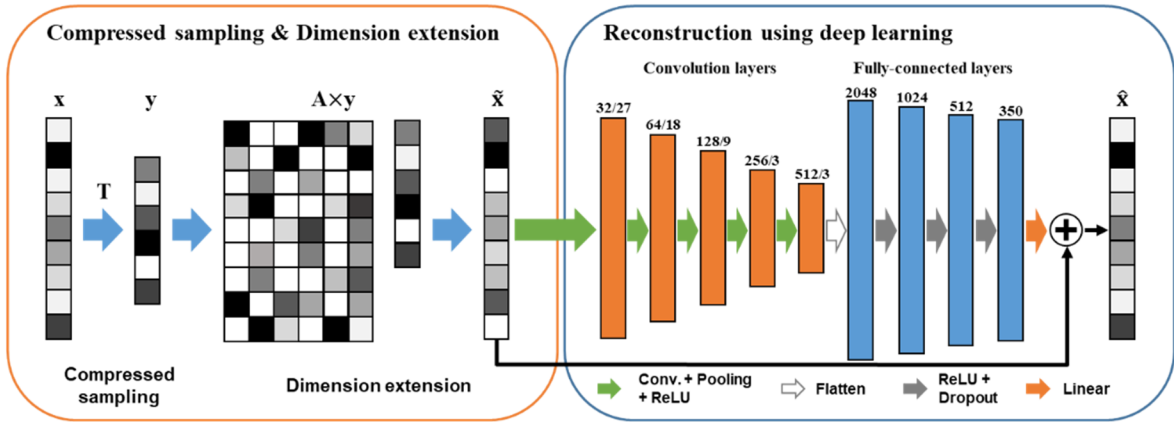


Figure 4.13 Overview of computational spectroscopy system including the proposed RCNN: An input spectrum is compressively sampled by the sensing matrix, and the dimension of measurements is extended by the transform matrix. RCNN is trained to recover the original spectrum from the extended measurements.

As depicted in Fig. 4.13, RCNN comprises nine learnable layers, five of which are convolution layers, four are fully-connected layers, and one is a residual connection. Convolution layers are used for the feature extraction in the non-linear mapping between $\tilde{\mathbf{x}}$ and \mathbf{x} . Fully-connected layers are used for the spectra reconstruction. Each of the convolutional layer has a set of one-dimensional learnable kernels with specific window sizes. The number of kernels and the window sizes are indicated in Fig. 4.13. After every convolutional layer, ReLU is used as an activation function, and the subsampling (Max pooling) is then applied. We stack the convolutional layer, the ReLU, and the subsampling five times. The output of the last subsampling is flattened and then fed into the subsequent four fully-connected layers. The first three layers are followed by the ReLU and dropout in sequence. The dropout is introduced to reduce the overfitting of RCNN. The output of the last fully-connected layer is fed into a linear activation function. The number of units in

each of the fully-connected layers is noted in Fig. 4.13. Moreover, we make the residual connection that the representative spectrum $\tilde{\mathbf{x}}$ and the output of the linear activation function are added up to the reconstructed spectrum $\hat{\mathbf{x}}$. Consequently, $\hat{\mathbf{x}}$ is trained to become \mathbf{x} . Given training data $\{\mathbf{x}_t^i\}_{i=1}^k$, we train RCNN to minimize a loss function L . We use the mean squared error between the original \mathbf{x}_t and recovered $\hat{\mathbf{x}}_t$ as the loss function:

$$L = \frac{1}{k} \sum_{i=1}^k \|\mathbf{x}_t^i - \hat{\mathbf{x}}_t^i\|_2^2. \quad (9)$$

The non-linear mapping that $\tilde{\mathbf{x}}$ becomes \mathbf{x} can be defined as $H(\tilde{\mathbf{x}}) = \mathbf{x}$. Because of the residual connection in RCNN, $H(\tilde{\mathbf{x}})$ can be rewritten as $H(\tilde{\mathbf{x}}) = F(\tilde{\mathbf{x}}) + \tilde{\mathbf{x}}$, where $F(\tilde{\mathbf{x}})$ is the mapping of the learnable layers. The representative spectrum $\tilde{\mathbf{x}}$ is referenced by the residual connection, and then, $F(\tilde{\mathbf{x}}) = H(\tilde{\mathbf{x}}) - \tilde{\mathbf{x}}$. In particular, the mapping of $F(\tilde{\mathbf{x}})$ is called a residual mapping; therefore, the learnable layers learn the residual of \mathbf{x} and .

Figure 4.14 depicts the manner in which a spectrum is recovered in CNN and RCNN. The learnable layers of CNN directly reconstruct the spectrum from the representative spectrum $\tilde{\mathbf{x}}$. Alternatively, RCNN reconstructs the spectrum by passing the representative spectrum $\tilde{\mathbf{x}}$ through the residual connection shown in Fig 4.14(b). Consequently, the learnable layers of RCNN learn to reconstruct residuals.

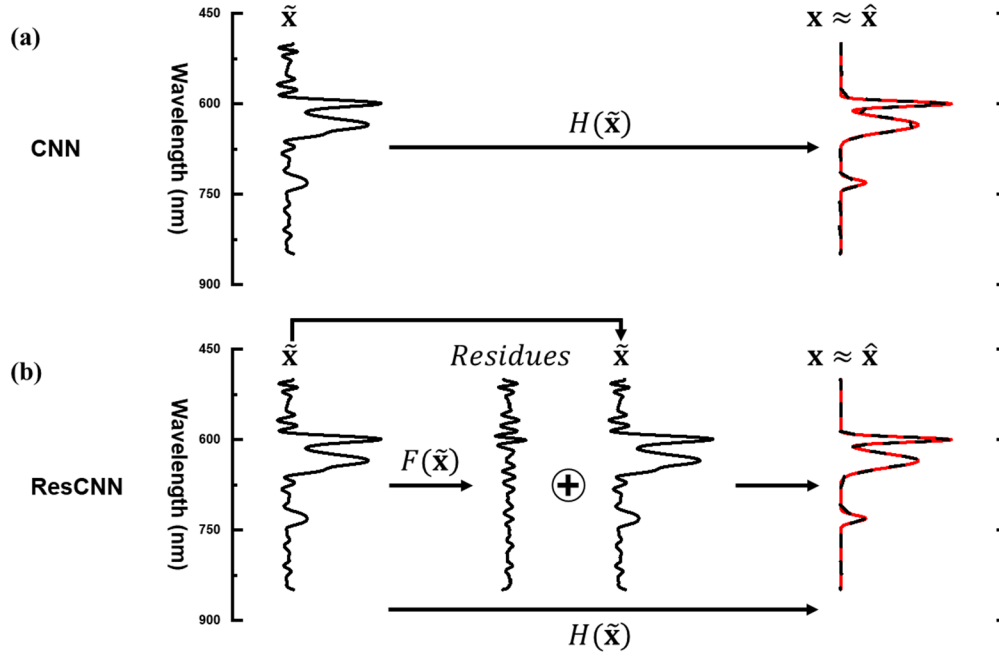


Figure 4.14 Descriptions of the spectrum recovery process: (a) CNN, (b) RCNN.

We reconstructed 350 spectral bands ($N = 350$) using 36 thin-film filters ($M = 36$) whose sensing patterns have a spacing of 1 nm for wavelengths from 500 to 850 nm. We determined the sensing matrix \mathbf{R} , assuming that the incident light falls onto the filters with normal incidence. As the transform matrix \mathbf{A} , we used the Moore-Penrose inverse of the sensing matrix \mathbf{R} , i.e., $\mathbf{A} = \mathbf{R}^T (\mathbf{R}\mathbf{R}^T)^{-1}$

4.3.3. Experimental Dataset

To evaluate the performance of RCNN, we used two synthetic spectral datasets and two measured spectral datasets. The first synthetic dataset is composed of Gaussian distribution functions while the other is composed of Lorentzian distribution functions. These two synthetic datasets were selected as generally these types of functions are used to represent spectral line shapes. As shown in Figure 4.15, component functions are added to produce the spectra. We generated 12,000 spectra for each dataset. For each spectrum, the number of component functions was generated using a geometric distribution with the probability parameter p set to 0.3. We added one to the number of component functions to prevent the number of component functions from becoming zero. Then, we randomly set a location, a

height, and an FWHM of each peak. To set a peak location (nm), an integer number is randomly selected from a uniform distribution with the interval (500, 849). A random number from a uniform distribution in the interval (0, 1) is used for the height. An integer number for an FWHM (nm) is randomly drawn from a uniform distribution with the interval (2, 50). Finally, all of the component functions are summed to generate the spectrum. The height of each generated spectrum is normalized such that it is mapped from zero to one.

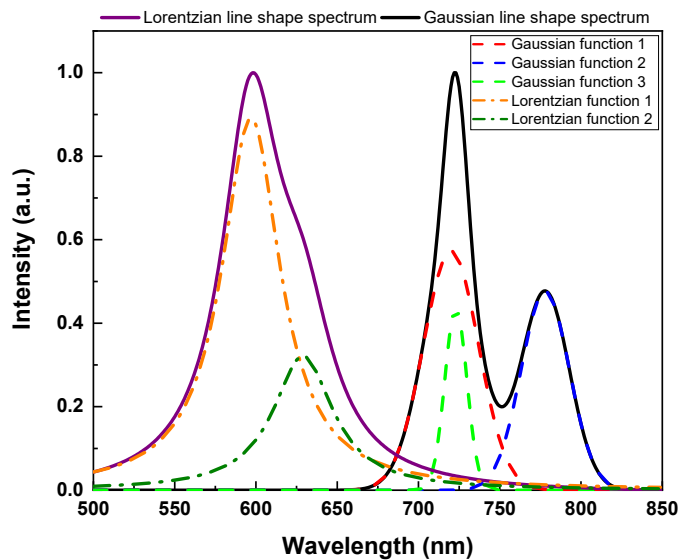


Figure 4.15 Examples of two synthetic spectra: the solid purple line represents a Lorentzian line shape spectrum and the solid black line represents a Gaussian line shape spectrum.

As measured datasets, we used the US Geological Survey (USGS) spectral library version 7 [59], and the glossy Munsell colors spectral dataset [60]. The USGS spectral library provides reflectance spectra for artificial materials, coatings, liquids, minerals, organic compounds, soil mixtures, and vegetation. We discarded any spectrum that has missing spectral bands. Then, we extracted the spectrum in the wavelength range of interest (500 to 849 nm) from the wavelength range of the original spectrum (350 to 2500 nm). The measured wavelength range for the glossy Munsell colors spectral dataset, which contains the reflectance spectra of the glossy Munsell color chips, is 380 to 780 nm. The wavelength range of the original spectrum is different from the wavelength range of

interest. We determined to use the wavelength range from 400 to 749 nm to ensure each spectrum was set to 350 spectral bands. This selection of wavelengths is reasonable because the wavelengths are located in the center of the wavelength range of the original spectrum, and show different spectral features with respect to each spectrum. In addition, our aim is to show the reconstruction performance with respect to various kinds of spectra. Finally, each spectrum is normalized such that the height varies from 0 to 1. Overall 1473 spectra from USGS spectral dataset and 1600 spectra from Munsell color spectral dataset were used for our simulated experiments. Table 4.3 lists the details of each of the spectral datasets.

Data Preprocessing and Training. Given the sensing matrix, the spectral data are compressively sampled as the measurement vector \mathbf{y} shown in Eq. (5), and then transformed into the representative spectrum $\tilde{\mathbf{x}}$ by multiplying the transform matrix \mathbf{A} and \mathbf{y} .

In each spectral dataset, the number of training, validation, and test spectra are randomly assigned using a ratio of 4:1:1 for the synthetic and measured data sets, respectively. The validation spectra are used for estimating the number of epochs and tuning the hyper-parameters. To train RCNN, we used the Adam optimizer [52] implemented in Tensorflow with a batch size of 16 and 250 epochs. The experiments were conducted on an NVIDIA GeForce RTX 2060 GPU. Training the architecture can be done in half an hour for each dataset.

Table 4.3 Description of the spectral datasets

Dataset	Training / Validation / Test	Avg. number of nonzero values	Description
Gaussian dataset	8000 / 2000 / 2000	336.8/350	FWHM (nm) on the interval [2, 50], Height on the interval [0, 1]
Lorentzian dataset	8000 / 2000 / 2000	349/350	FWHM (nm) on the interval [2, 50], Height on the interval [0, 1]
USGS [59]	982 / 246 / 245	348.9/350	350-2500 nm, 2151 spectral bands (we use 350 spectral bands in 500-849 nm)
Munsell colors [60]	1066 / 267 / 267	349/350	380-780 nm, 401 spectral bands (we use 350 spectral bands in 400-749 nm)

4.3.4. Dictionary Learning for Sparse Recovery

We evaluated the performance of conventional reconstructions using sparse recovery to benchmark the performance of RCNN. As shown in Table 4.3, the spectra for both the synthetic and measured datasets are dense spectra. Therefore, we must transform the spectra into sparse signals to solve Eq. (8). In this section, we considered methods to make a sparsifying basis Φ .

First, we considered a Gaussian line shape matrix as a sparsifying basis. Each column of the matrix comprises a Gaussian distribution function whose length is N . A collection of N Gaussian functions works as a sparsifying basis $\Phi \in \mathbb{R}^{N \times N}$. We generate two Gaussian line shape matrices. Figure 4.16(a) shows the heatmap images for two Gaussian line shape matrices. Seven different FWHMs are used to generate the Gaussian distributions. Given an FWHM, Gaussian distributions are generated by shifting the peak location using a uniform spacing. To create a small dissimilarity between the two Gaussian line shape matrices, two of the seven FWHMs in Gaussian 1 were replaced with other FWHMs, thus producing Gaussian 2, as shown in Fig. 4.16(a).

Second, learned dictionary [61]–[64] is used as a sparsifying basis. Given a training dataset $\{\mathbf{x}_t^i\}_{i=1}^k$, we can derive a learned dictionary Φ that sparsely represents the training data \mathbf{x}_t by solving the following optimization problem, known as the dictionary learning problem:

$$\min_{\Phi, \mathbf{s}_t^1, \dots, \mathbf{s}_t^k} \sum_{i=1}^k \|\mathbf{x}_t^i - \Phi \mathbf{s}_t^i\|_2^2 + \tau \|\mathbf{s}_t^i\|_1, \quad (10)$$

where τ is a regularization parameter and \mathbf{s}_t^i is i^{th} sparse signal over the training dataset. By fixing an initial guess for the dictionary Φ in Eq. (10), we obtain a solution for the sparse signals $\{\mathbf{s}_t^i\}_{i=1}^k$. The dictionary is then updated by solving Eq. (10) using the obtained sparse signals. This process is iteratively repeated until convergence is reached and we derive the learned dictionary. We used three dictionary learning methods: method of optimal directions (MOD) [62], K-SVD [63], and online dictionary learning (ODL) [64]. The learned dictionaries are generated for each of the training datasets, and the

reconstruction performances are evaluated for each test dataset. Figure 4.16(b) shows learned dictionaries identified using the Gaussian training dataset. The learned dictionaries clearly depend on the dictionary learning methods used. Nevertheless, each column of the dictionaries shows a learned spectral feature from the training dataset.

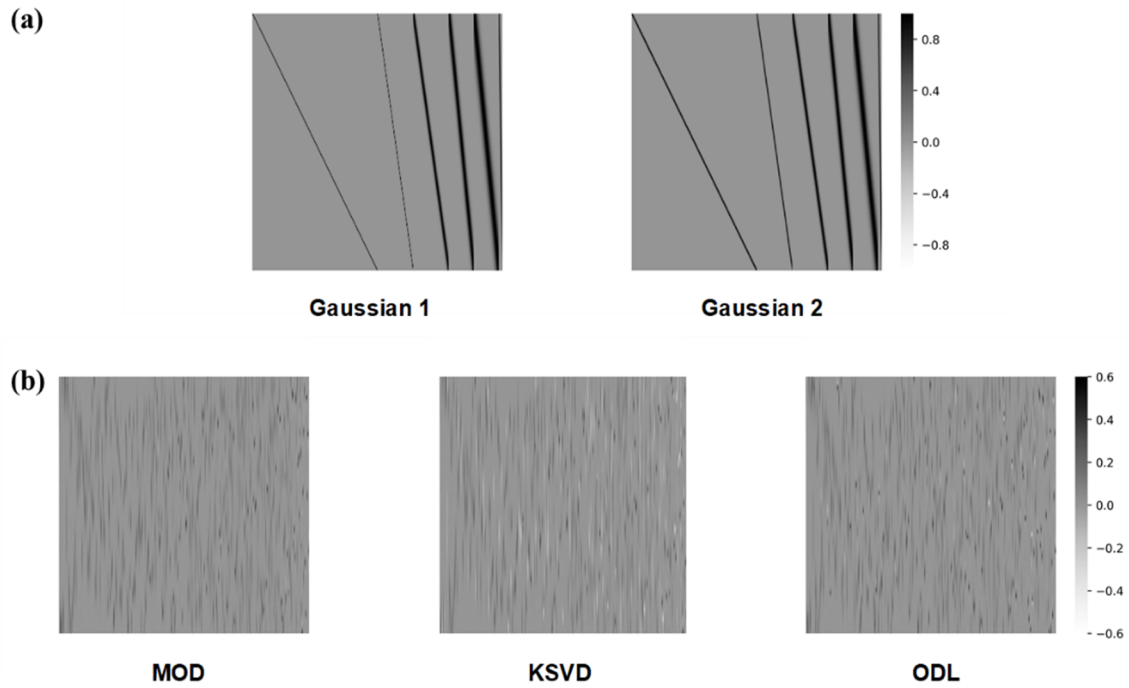


Figure 4.16 Sparsifying bases for the sparse recovery: (a) Gaussian line shape matrices and (b) learned dictionaries both from the Gaussian training dataset.

4.3.5. Results

To demonstrate the ability of RCNN to reconstruct spectra, we evaluated its performance using three different datasets: synthetic datasets, noisy synthetic datasets, and measured datasets. We used the same hyper-parameters of RCNN for each of these datasets. Moreover, we adopted $l1_ls$ [38] as the fixed reconstruction algorithm in the sparse recovery. We compared the recovered signal with the original signal by calculating the root mean squared error (RMSE) and the peak signal to noise ratio (PSNR). In addition, the performances of five conventional sparse recovery methods and CNN were calculated.

The two synthetic data sets described in Table 4.3 were used to perform the signal recovery using sparse recovery and deep learning. Table 4.4 shows the average RMSE and PSNR for each of the seven methods evaluated. RCNN shows the smallest average RMSE for both the Gaussian and Lorentzian datasets of 0.0094 and 0.0073, respectively. Moreover, RCNN shows the largest average PSNR of 49.0 dB for the Lorentzian dataset. For the Gaussian dataset, the sparse recovery method with Gaussian 2 shows the largest average PSNR, 49.7 dB, which is slightly higher than the 47.2 dB for RCNN. Note that the minor difference between the two Gaussian line shape matrices results in considerable performance differences. However, reconstructions using the learned dictionaries show similar performance across all of the synthetic datasets.

Table 4.4 Avg. RMSEs and PSNRs over synthetic datasets

Dataset	Sparse recovery				Deep learning		
	Gaussian 1	Gaussian 2	K-SVD	MOD	ODL	CNN [18]	RCNN [14]
Gaussian dataset	0.0226 (43.1 dB)	0.0112 (49.7 dB)	0.0172 (40.3 dB)	0.0174 (40.3 dB)	0.0161 (41.1 dB)	0.0132 (40.5 dB)	0.0094 (47.2 dB)
Lorentzian dataset	0.0146 (44.9 dB)	0.0094 (47.5 dB)	0.0136 (42.3 dB)	0.0137 (42.3 dB)	0.0127 (42.9 dB)	0.0101 (42.8 dB)	0.0073 (49.0 dB)

Figure 4.17 shows the reconstructed test spectra from each of the synthetic datasets. The solid red lines (i) are the input spectra from each dataset. RCNN is shown in dashed black lines (ii), while CNN is shown in solid orange lines (iii). The reconstructed spectra using sparse recovery with Gaussian 1 (iv), Gaussian 2 (v), and ODL (vi) are shown in solid green, blue, and purple lines in respectively. Because of the similar performance from each of the learned dictionaries only the ODL method is shown. The RMSE and PSNR of RCNN are 0.0138 (37.2 dB) for the spectrum from the Gaussian dataset and 0.0096 (40.4 dB) for the spectrum from the Lorentzian dataset. For the selected spectra, RCNN achieves superior reconstruction performance compared with the other four reconstructions.

Only sparse recovery with Gaussian 1 fails to recover the fine details of the input spectrum. One example of the poor ability of sparse recovery with Gaussian 1 to resolve the signal is the recovery of the peak at ~ 830 and 590 nm being recovered as two neighboring peaks in Figure 4.17(a) and (b), respectively. CNN was unable to capture the smoothness of the spectral features compared to the other methods.

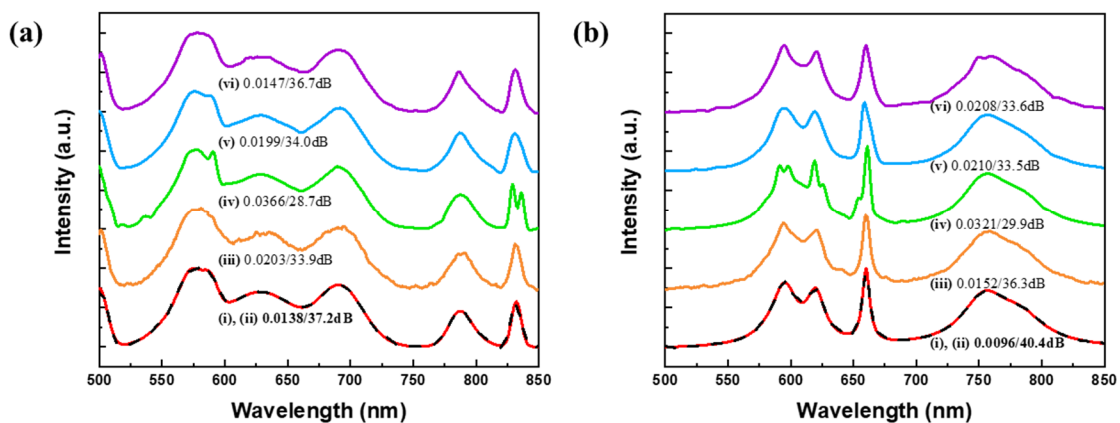


Figure 4.17 Spectral reconstructions of test spectra in synthetic datasets, (a) Gaussian dataset, (b) Lorentzian dataset. An input spectrum (solid red (i)) is compared with RCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and ODL (purple (vi)). The baselines are shifted for clarity.

RCNN was trained using the two measured datasets listed in Table 4.3, USGS and Munsell colors, and its reconstruction performance was evaluated. In addition, the signal reconstruction was performed using CNN and sparse recovery with five different sparsifying bases. Table 4.5 reports the average RMSE and PSNR for each of the seven methods. RCNN achieves the smallest average RMSE and the largest average PSNR for both datasets. In USGS dataset, the average RMSE and PSNR of RCNN are 0.0048 and 52.4 dB, respectively. In addition, RCNN achieves 0.0040 for the average RMSE and 50.0 dB for the average PSNR in Munsell colors dataset. Similar to synthetic datasets, all of the learned dictionaries provided similar reconstruction performances. In addition, the small differences between Gaussian 1 and 2 show large differences in the RMSE and PSNR. The average RMSE and PSNR of the learned dictionary methods approach the values of RCNN for Munsell colors dataset because Munsell colors dataset has simpler spectral features than the other datasets.

Table 4.5 Avg. RMSEs and PSNRs for the measured datasets

Dataset	Sparse recovery					Deep learning	
	Gaussian 1	Gaussian 2	K-SVD	MOD	ODL	CNN [18]	RCNN [14]
USGS [59]	0.0081 (45.3 dB)	0.0061 (48.4 dB)	0.0070 (48.5 dB)	0.0081 (47.4 dB)	0.0074 (47.6 dB)	0.0116 (40.8 dB)	0.0048 (52.4 dB)
Munsell colors [60]	0.0068 (44.6 dB)	0.0050 (47.5 dB)	0.0040 (49.8 dB)	0.0040 (49.9 dB)	0.0042 (49.5 dB)	0.0076 (43.0 dB)	0.0040 (50.0 dB)

Figure 4.18 shows the reconstruction results of one test spectra from each of the measured datasets. The spectrum for the organic compound dibenzothiophene in the USGS dataset is reconstructed in Figure 4.18(a). The spectrum of Munsell color 5 PB 2/2 is shown in Figure 4.18(b). The solid red lines are the input spectra (i). The results of RCNN are shown in dashed black lines (ii), and The results of CNN are shown in solid black lines (iii). The spectra of (iv) to (vi) are reconstructed spectra using the sparse recovery with Gaussian 1, Gaussian 2, and K-SVD. Because of the best performance of the K-SVD among the learned dictionaries only the K-SVD method is shown.

The RMSE and PSNR for RCNN are 0.0069 (43.2 dB) for the spectrum from USGS dataset and 0.0077 (42.3 dB) for the spectrum from Munsell colors dataset. RCNN outperforms other approaches for the spectrum from USGS dataset. However, for the spectrum from Munsell colors dataset, the sparse recovery with K-SVD outperforms RCNN. RCNN achieves slightly larger RMSE and smaller PSNR.

The performances of sparse recovery with Gaussian 2 is degraded for measured datasets compared with the performances for synthetic datasets. The measured datasets have rough spectral features unlike the smooth spectral features observed in the synthetic datasets. As a result, the sparse recovery with Gaussian 2 performs worse, because of its inability to represent rough spectral features using Gaussian distribution functions. The performances of sparse recovery with dictionary learning methods are improved for measured datasets compared with the performances of synthetic datasets. Because the number of spectra in measured datasets are smaller than the number of spectra in synthetic datasets. Therefore, finding the best-fit sparsifying basis for measured datasets is easier than finding the best-fit sparsifying basis for synthetic datasets using dictionary learning methods. Meanwhile, RCNN shows superior reconstruction performances regardless of the spectral features of datasets and the size of datasets.

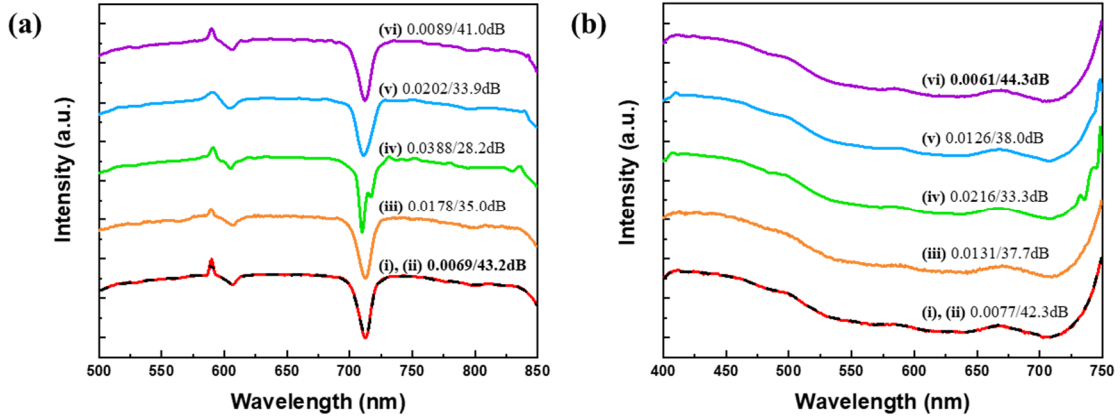


Figure 4.18 Spectral reconstructions of test spectra in measured datasets: (a) spectrum of organic compound dibenzothiophene in USGS dataset, (b) spectrum of Munsell color 5PB 2/2. The input spectrum (solid red line (i)) is compared with RCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and K-SVD (purple (vi)). The baselines are shifted for clarity.

To verify the stability of RCNN, we evaluated the accuracy of the reconstruction at various noise levels. Gaussian white noise was added to the measurement vector $\mathbf{n} \in \mathbb{R}^{M \times 1}$ to Eq. (5). We considered six different noise levels whose signal-to-noise ratios (SNRs) are 15, 20, 25, 30, 35, and 40 dB. Using Gaussian and Lorentzian datasets, we compared the reconstruction performance of RCNN with the sparse recovery using Gaussian 2, which shows the best reconstruction performances among sparse recovery methods in synthetic datasets. RCNN was evaluated with the same hyper-parameters that were used for the noise-free datasets. The average RMSE and PSNR for each of the six noise levels are shown in Table 4.6. While RCNN was trained using noise-free data, it outperformed the sparse recovery with Gaussian 2 at every noise level, which indicates that RCNN remains stable even with the noisy dataset

Table 4. 6 Avg.RMSE and PSNR under various SNR (dB) with synthetic datasets

		SNR (dB)					
Dataset	Method	15 dB	20 dB	25 dB	30 dB	35 dB	40 dB
Gaussian dataset	Sparse recovery +	0.0796	0.0482	0.0308	0.0215	0.0166	0.0138
	Gaussian 2	(22.7 dB)	(27.1 dB)	(31.2 dB)	(34.8 dB)	(37.9 dB)	(40.7 dB)
	RCNN	0.0671 (24.2 dB)	0.0401 (28.7 dB)	0.0251 (32.9 dB)	0.0171 (36.6 dB)	0.0130 (39.8 dB)	0.0110 (42.4 dB)
Lorentzian dataset	Sparse recovery +	0.0817	0.0483	0.0300	0.0201	0.0147	0.0119
	Gaussian 2	(22.6 dB)	(27.1 dB)	(31.2 dB)	(35.0 dB)	(38.5 dB)	(41.4 dB)
	RCNN	0.0689 (24.1 dB)	0.0404 (28.7 dB)	0.0243 (33.1 dB)	0.0157 (37.1 dB)	0.0113 (40.6 dB)	0.0091 (43.4 dB)

4.3.6. Summary

As shown in the results, we empirically demonstrate that RCNN outperforms the sparse recovery methods and the CNN over all datasets. The sparse recovery shows unstable performance because it is highly dependent on the sparsifying basis and spectral features of the dataset. This is a direct result of being unable to identify a fixed sparsifying basis that can transform any spectra into a sparse signal, which means the *a priori* structural information such as line shapes and FWHMs are required to select a consistent sparsifying basis. Learned dictionaries are used to cope with the problem of identifying a consistent sparsifying basis. The columns of learned dictionaries are composed of learned spectral features from the training dataset. While this shows an improvement in measured datasets, a learned dictionary is still limited to represent all the spectral features in the large dataset (i.e., synthetic datasets) using linear combinations of columns of the learned dictionary. For recovering spectra, RCNN does not require the *a priori* knowledge of a sparsifying basis or prior information of spectral features. During training, RCNN learns the spectral features using learnable layers, which enable it to recover the fine details for various kinds of spectra without identifying a sparsifying basis.

RCNN is directly compared with CNN for the synthetic Gaussian dataset in Fig. 4.19(a) where the mean squared error is plotted with respect to the epoch. The mean squared error for CNN and RCNN are shown in solid black line and solid red line with square symbols, respectively. RCNN shows a lower mean squared error than that of CNN. Moreover, RCNN converges faster than CNN, indicating that RCNN optimizes the learnable layers quicker, as expected based on previous research using residuals [41], [51]. In contrast to the previous research that multiple residual connections were used in very deep neural networks to converge faster by avoiding the vanishing gradient problem, we achieve spectral reconstruction improvements even with one residual connection in moderate depth CNN.

The reconstruction of an example spectrum with respect to the number of epochs is shown in Fig. 4.19(b). Black lines ((i) to (iv)) are the reconstructed spectra at 1, 50, 150, and 250 epochs, respectively. The solid red line (v) is the original spectrum, and the series of reconstructed spectra for RCNN show that the reconstruction converged earlier than CNN.

The increased rate of convergence is because of the residual connection in RCNN. Overall the reconstruction performance of RCNN is an improvement over CNN.

Note that both RCNN and dictionary learning for sparse recovery require a training dataset and an optimization process to learn the spectral features. While this is a time-consuming process, remember that when using a learned dictionary to recover spectra, an iterative reconstruction algorithm is required, which needs additional time and takes a large computational cost. The benefit of RCNN is that it gives a reconstructed spectrum immediately once the training is completed.

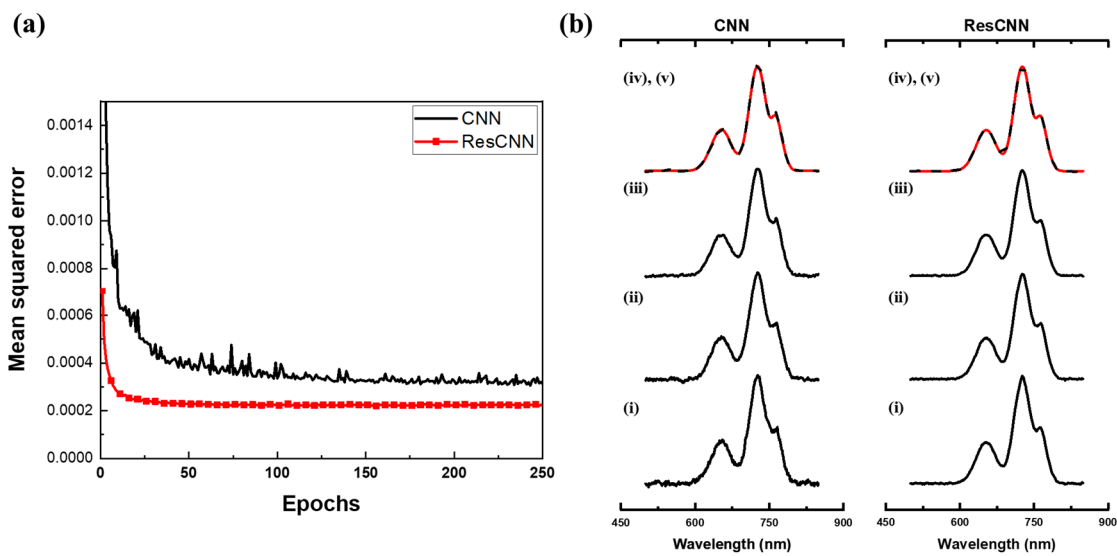


Figure 4.19 (a) Mean squared error of Gaussian dataset with respect to epochs. Solid black line denotes validation error of CNN, and solid red line with square symbols denotes validation error of RCNN. (b) Reconstructions of a spectrum with respect to epochs where (i) to (iv) are epochs 1, 50, 150, and 250, respectively. Red line (v) denotes the original spectrum.

Chapter 5

Conclusions

In this dissertation, we demonstrate MTF filter-based computational spectrometers. The computational spectrometers have a compact size, a high resolution, a wide working range, and a fast operation time. MTF filters are designed based on the compressed sensing theory. We propose a mass production enable fabrication process for fabricating MTF filter arrays. We build a spectrometer by attaching the fabricated MTF filter array to a CMOS image sensor. The spectrometer is compact and lightweight.

Using the fabricated MTF filter-based spectrometers, we measure the intensities of an input light source into a digital signal. From the measured intensities, we reconstruct the spectra of the input light in a high resolution using computational approaches. As the computational approaches, we use sparse recovery based on numerical optimization, and a residual convolutional neural network (RCNN) based on deep learning. We formulate the problem of the underdetermined linear system for spectrometers and solve the problem using these computational approaches.

We conduct numerical experiments for estimating the performances of the computational spectrometer. After numerical experiments, we perform optical experiments using various light sources, including monochromatic lights, LEDs, and a halogen lamp. The results show good recovery performances over the light sources. In addition, we compare spectral reconstruction performances between sparse recovery and RCNN over four kinds of datasets and noisy environments. Although there were limitations in showing the generalization of deep learning architectures, RCNN shows better reconstruction than the sparse recovery over all datasets.

The demonstration of the MTF filter arrays for computational spectrometers is an important step towards the industrialization and practical uses of computational

spectrometers. This study will be helpful for computational spectroscopy to be realized in various applications.

Finally, one advantage of filter-based spectrometers is that they could offer spatial information. Using a microlens array, and a separator, we could implement non-scanning spectral imaging systems. We assume that it is possible to construct a computational snapshot spectral imaging system combining the fabricated MTF filter array, a microlens array, and a separator. As a future research direction, we will focus on computational spectral imaging using the MTF filter array.

Bibliography

- [1] N. V. Tkachenko, *Optical spectroscopy: methods and instrumentations*. Elsevier, 2006.
- [2] P. D. Atherton, K. Taylor, C. D. Pike, C. F. Harmer, N. M. Parker, and R. N. Hook, “TAURUS: a wide-field imaging Fabry–Perot spectrometer for astronomy,” *Monthly Notices of the Royal Astronomical Society*, vol. 201, no. 3, pp. 661–696, 1982.
- [3] B. Henderson and G. F. Imbusch, *Optical spectroscopy of inorganic solids*, vol. 44. Oxford University Press, 2006.
- [4] R. N. Clark and T. L. Roush, “Reflectance spectroscopy: Quantitative analysis techniques for remote sensing applications,” *Journal of Geophysical Research: Solid Earth*, vol. 89, no. B7, pp. 6329–6340, 1984.
- [5] D. L. Donoho, “Compressed sensing,” *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [6] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, “Compressed sensing with coherent and redundant dictionaries,” *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59–73, 2011.
- [7] E. J. Candes and T. Tao, “Decoding by linear programming,” *IEEE transactions on information theory*, vol. 51, no. 12, pp. 4203–4215, 2005.
- [8] O. Lee, J. M. Kim, Y. Bresler, and J. C. Ye, “Compressive diffuse optical tomography: noniterative exact reconstruction using joint sparsity,” *IEEE transactions on medical imaging*, vol. 30, no. 5, pp. 1129–1142, 2011.
- [9] A. Wagadarikar, R. John, R. Willett, and D. Brady, “Single disperser design for coded aperture snapshot spectral imaging,” *Applied optics*, vol. 47, no. 10, pp. B44–B51, 2008.
- [10] M. A. Herman and T. Strohmer, “High-resolution radar via compressed sensing,” *IEEE transactions on signal processing*, vol. 57, no. 6, pp. 2275–2284, 2009.
- [11] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, p. 436, 2015.
- [12] J. Bouvrie, “Notes on convolutional neural networks,” 2006.

- [13] C. Kim, W.-B. Lee, S. K. Lee, Y. T. Lee, and H.-N. Lee, "Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy," *Optics and Lasers in Engineering*, vol. 115, pp. 53–58, 2019.
- [14] C. Kim, D. Park, and H.-N. Lee, "Compressive sensing spectroscopy using a residual convolutional neural network," *Sensors*, vol. 20, no. 3, p. 594, 2020.
- [15] Cheolsun Kim, Pavel Ni, Kang Ryeol Lee, and Heung-No Lee, "Mass production-enabled computational spectrometer based on multilayer thin films," in revision.
- [16] C. Kim, D. Park, and H.-N. Lee, "Deep learning based single-shot computational spectrometer using multilayer thin films," in preparation.
- [17] W.-B. Lee, C. Kim, G. W. Ju, Y. T. Lee, and H.-N. Lee, "Design of thin-film filters for resolution improvements in filter-array based spectrometers using dsp," in *Next-Generation Spectroscopic Technologies IX*, 2016, vol. 9855, p. 98550Q.
- [18] C. Kim, D. Park, and H.-N. Lee, "Convolutional neural networks for the reconstruction of spectra in compressive sensing spectrometers," in *Optical Data Science II*, 2019, vol. 10937, p. 109370L.
- [19] C. P. Bacon, Y. Mattley, and R. DeFrece, "Miniature spectroscopic instrumentation: applications to biology and chemistry," *Review of Scientific Instruments*, vol. 75, no. 1, pp. 1–16, 2004.
- [20] S. Kim *et al.*, "Smartphone-based multispectral imaging: system development and potential for mobile skin diagnosis," *Biomedical optics express*, vol. 7, no. 12, pp. 5294–5307, 2016.
- [21] U. Kurokawa, B. I. Choi, and C.-C. Chang, "Filter-based miniature spectrometers: spectrum reconstruction using adaptive regularization," *IEEE Sensors Journal*, vol. 11, no. 7, pp. 1556–1563, 2011.
- [22] Z. Yang, T. Albrow-Owen, W. Cai, and T. Hasan, "Miniaturization of optical spectrometers," *Science*, vol. 371, no. 6528, 2021.
- [23] J. Oliver, W. Lee, S. Park, and H.-N. Lee, "Improving resolution of miniature spectrometers by exploiting sparse nature of signals," *Optics express*, vol. 20, no. 3, pp. 2613–2625, 2012.
- [24] J. Oliver, W.-B. Lee, and H.-N. Lee, "Filters with random transmittance for improving resolution in filter-array-based spectrometers," *Optics express*, vol. 21, no. 4, pp. 3969–3989, 2013.

- [25] Z. Wang and Z. Yu, "Spectral analysis based on compressive sensing in nanophotonic structures," *Optics express*, vol. 22, no. 21, pp. 25608–25614, 2014.
- [26] J. Bao and M. G. Bawendi, "A colloidal quantum dot spectrometer," *Nature*, vol. 523, no. 7558, pp. 67–70, 2015.
- [27] E. Huang, Q. Ma, and Z. Liu, "Etalon Array Reconstructive Spectrometry," *Scientific reports*, vol. 7, 2017, Accessed: Sep. 10, 2017. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5225422/>
- [28] B. Cerjan and N. J. Halas, "Toward a Nanophotonic Nose: A Compressive Sensing-Enhanced, Optoelectronic Mid-Infrared Spectrometer," *ACS Photonics*, vol. 6, no. 1, pp. 79–86, 2018.
- [29] Z. Wang *et al.*, "Single-shot on-chip spectral sensors based on photonic crystal slabs," *Nature communications*, vol. 10, no. 1, p. 1020, 2019.
- [30] Z. Yang *et al.*, "Single-nanowire spectrometers," *Science*, vol. 365, no. 6457, pp. 1017–1020, 2019.
- [31] Y. Kwak, S. M. Park, Z. Ku, A. Urbas, and Y. L. Kim, "A Pearl Spectrometer," *Nano Letters*, 2020.
- [32] Y. Oiknine, I. August, D. G. Blumberg, and A. Stern, "Compressive sensing resonator spectroscopy," *Optics letters*, vol. 42, no. 1, pp. 25–28, 2017.
- [33] Y. Zhu, X. Lei, K. X. Wang, and Z. Yu, "Compact CMOS spectral sensor for the visible spectrum," *Photonics Research*, vol. 7, no. 9, pp. 961–966, 2019.
- [34] D. Gedalin, Y. Oiknine, and A. Stern, "DeepCubeNet: reconstruction of spectrally compressive sensed hyperspectral images with deep neural networks," *Optics express*, vol. 27, no. 24, pp. 35811–35822, 2019.
- [35] J. Zhang, X. Zhu, and J. Bao, "Solver-informed neural networks for spectrum reconstruction of colloidal quantum dot spectrometers," *Optics Express*, vol. 28, no. 22, pp. 33656–33672, 2020.
- [36] C. Brown *et al.*, "Neural network-based on-chip spectroscopy using a scalable plasmonic encoder," *ACS nano*, vol. 15, no. 4, pp. 6305–6315, 2021.
- [37] W. Zhang *et al.*, "Deeply learned broadband encoding stochastic hyperspectral imaging," *Light: Science & Applications*, vol. 10, no. 1, pp. 1–7, 2021.

- [38] K. Koh, S.-J. Kim, and S. Boyd, “An interior-point method for large-scale l_1 -regularized logistic regression,” *Journal of Machine learning research*, vol. 8, no. Jul, pp. 1519–1555, 2007.
- [39] J. M. Bioucas-Dias and M. A. Figueiredo, “A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration,” *IEEE Transactions on Image processing*, vol. 16, no. 12, pp. 2992–3004, 2007.
- [40] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Transactions on signal processing*, vol. 57, no. 7, pp. 2479–2493, 2009.
- [41] D. Lee, J. Yoo, and J. C. Ye, “Deep residual learning for compressed sensing MRI,” in *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, 2017, pp. 15–18.
- [42] H. A. Macleod, *Thin-film optical filters*. CRC press, 2001. Accessed: Sep. 11, 2017. [Online]. Available: <https://books.google.co.kr/books?hl=ko&lr=&id=D0S9hxxzPJq8C&oi=fnd&pg=PR13&dq=Thin-Film+Optical+Filters&ots=t69TmeGMHs&sig=e8Uwhc2IPo7LFNyM4q-vsGNhkfk>
- [43] J. R. Barry and J. M. Kahn, “Link design for nondirected wireless infrared communications,” *Applied optics*, vol. 34, no. 19, pp. 3764–3776, 1995.
- [44] F. L. Pedrotti and L. S. Pedrotti, “Introduction to optics 2nd edition,” *Introduction to Optics 2nd Edition by Frank L. Pedrotti, SJ, Leno S. Pedrotti New Jersey: Prentice Hall, 1993*, 1993, Accessed: Sep. 11, 2017. [Online]. Available: <http://adsabs.harvard.edu/abs/1993io2..book.....P>
- [45] D. M. Topasna and G. A. Topasna, “Numerical modeling of thin film optical filters,” in *Proc. of SPIE Vol*, 2009, vol. 9666, pp. 96661P–1. Accessed: Sep. 11, 2017. [Online]. Available: <http://proceedings.spiedigitallibrary.org/pdfaccess.ashx?url=/data/conferences/spiep/85062/96661p.pdf>
- [46] S.-W. Wang, M. Li, C.-S. Xia, H.-Q. Wang, X.-S. Chen, and W. Lu, “128 channels of integrated filter array rapidly fabricated by using the combinatorial deposition technique,” *Applied Physics B*, vol. 88, no. 2, pp. 281–284, 2007.

- [47] C. Liu and Z. Sun, “Design and fabrication of a metallic irregular F–P filter array for a miniature spectrometer,” *Applied Optics*, vol. 60, no. 16, pp. 4948–4953, 2021.
- [48] J. Tanida *et al.*, “Thin observation module by bound optics (TOMBO): concept and experimental verification,” *Applied optics*, vol. 40, no. 11, pp. 1806–1813, 2001.
- [49] N. Gupta, P. R. Ashe, and S. Tan, “Miniature snapshot multispectral imager,” *Optical Engineering*, vol. 50, no. 3, p. 033203, 2011.
- [50] B. Geelen, N. Tack, and A. Lambrechts, “A snapshot multispectral imager with integrated tiled filters and optical duplication,” in *Advanced Fabrication Technologies for Micro/Nano Optics and Photonics VI*, 2013, vol. 8613, p. 861314.
- [51] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [52] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv pre-print arXiv:1412.6980*, 2014.
- [53] “vitaminwater,” *official vitaminwater website*. <https://www.vitaminwater.com/home> (accessed Aug. 19, 2021).
- [54] “POWERADE® Sports Drink | More Power For You,” *POWERADE - The Official Website for Powerade*. <https://www.powerade.com/> (accessed Aug. 19, 2021).
- [55] A. Mousavi and R. G. Baraniuk, “Learning to invert: Signal recovery via deep convolutional networks,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2017, pp. 2272–2276.
- [56] Y. Li, Y. Xue, and L. Tian, “Deep speckle correlation: a deep learning approach toward scalable imaging through scattering media,” *Optica*, vol. 5, no. 10, pp. 1181–1190, 2018.
- [57] M. Mardani *et al.*, “Deep Generative Adversarial Neural Networks for Compressive Sensing MRI,” *IEEE Transactions on Medical Imaging*, vol. 38, no. 1, pp. 167–179, Jan. 2019, doi: 10.1109/TMI.2018.2858752.
- [58] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, “Deep convolutional neural network for inverse problems in imaging,” *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4509–4522, 2017.
- [59] R. F. Kokaly *et al.*, *USGS Spectral Library Version 7 Data: US Geological Survey data release*. 2017.

- [60] “University of Eastern Finland, Spectral Color Research Group, <http://www.uef.fi/web/spectral/-spectral-database>.”
- [61] G. Chen and D. Needell, “Compressed sensing and dictionary learning,” *Finite Frame Theory: A Complete Introduction to Overcompleteness*, vol. 73, p. 201, 2016.
- [62] K. Engan, S. O. Aase, and J. H. Husoy, “Method of optimal directions for frame design,” in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, 1999, vol. 5, pp. 2443–2446.
- [63] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [64] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online dictionary learning for sparse coding,” in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 689–696.

Curriculum Vitae

Name : Cheolsun Kim
Birth Date : March. 15. 1993.
Birth Place : Republic of Korea
Permanent Address : 123 Cheomdangwagi-ro (Oryong-dong), Buk-gu, Gwangju
E-mail : csk0315@gist.ackr, feseen2@gmail.com

Research Interests

1. Computational Spectroscopy
2. Compressed Sensing / Machine Learning
3. Spectral Imaging

Education

2016.3 – 2022.2 School of Electrical Engineering and Computer Science,
Gwangju Institute of Science and Technology (Ph.D.)
2012.3 – 2016.2 School of Electrical Engineering and Computer Science,
Gwangju Institute of Science and Technology (B.S.)

Patent

1. Heung-No Lee, Woong-Bi Lee, **Cheolsun Kim**, “Image sensing apparatus, image sensing method, distance measuring apparatus, and distance measuring method”, patent number: US10,484,585B2, Date of patent: Nov. 19, 2019
2. 이흥노, 이용비, **김철순**, “촬상장치, 촬상방법, 거리측정장치, 및 거리측정방법”, 특허 번호: 10-1865126, 등록일: 2018. 05. 31, 출원 번호: 10-2016-0099298, 출원일: 2016. 08. 04.
3. **김철순**, 이흥노, 이용비, “하이퍼스펙트럴 이미지 장치”, 특허 번호: 10-1986998, 등록일: 2019. 06. 03, 출원 번호: 10-2017-0077480, 출원일: 2017. 06. 19.
4. 이흥노, **김철순**, 이용비, 이수경, 이용탁, 주건우 “분광장치 및 분광방법”, 특허 번호: 10-2030735, 등록일: 2019. 10. 02, 출원 번호: 10-2018-0010046, 출원일: 2018. 01. 26.

Awards

1. 광주과학기술원 전기전자컴퓨터공학부, Research Assistant 우수 연구상, 2019, 2020 년.

International Journal Papers (SCI / SCIE)

1. **Cheolsun Kim**, Dongju Park, and Heung-No Lee*, “Compressive sensing spectroscopy using a residual convolutional neural network”, MDPI Sensors, Vol. 20(3), Jan. 2020.
2. **Cheolsun Kim**, Woong-Bi Lee, Soo Kyung Lee, Yong Tak Lee, and Heung-No Lee*, “Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy”, Optics and Lasers in Engineering, Vol. 115, pp. 53-58, Apr. 2019.

International Conference

1. Woong-Bi Lee, **Cheolsun Kim**, Gun Wu Ju, Yong Tak Lee, and Heung-No Lee, “Design of thin-film filters for resolution improvements in filter-array based spectrometers using DSP”, SPIE Defense + Commercial Sensing 2016, Baltimore, USA, Apr. 17-21, 2016.
2. **Cheolsun Kim**, Woong-Bi Lee, Gun Wu Ju, Jeonghoon Cho, Seongmin Kim, Jinkyung Oh, Dongsung Lim, Yong Tak Lee, and Heung-No Lee “A method of incident angle estimation for high resolution spectral recovery in filter-array-based spectrometers,” Proc. SPIE 10117, San Francisco, US, Jan. 27-Feb.1, 2017.
3. **Cheolsun Kim**, Dongju Park, and Heung-No Lee, “Convolutional neural networks for the reconstruction of spectra in compressive sensing spectrometers”, SPIE Photonics West 2019, San Francisco, USA, Feb. 2-9, 2019.

Acknowledgement

2012 년 3 월 학사과정을 시작하여 2022 년 2 월 박사학위 과정을 마치기 까지 광주과학기술원에서의 약 10 년간의 생활은 행운이었습니다. 학사 과정 동안 수준 높은 교육과 해외 교류의 기회를 받아 연구자가 될 식견과 견문을 넓힐 수 있었습니다. 또한, 박사학위 과정을 통해서 학문에 대한 접근 방법과 문제 해결능력을 기를 수 있었습니다. 무엇보다도 이러한 소중한 기회와 자산을 가족과 애인 곁에서 누릴 수 있어 행운이었습니다. 또한, 여러 고마운 분들의 도움과 격려, 지지와 성원을 통해 박사학위 과정을 무사히 끝마칠 수 있었다고 생각하며 이에 진심으로 감사드립니다.

먼저, 저의 지도교수님 이흥노 교수님께 진심으로 감사드립니다. 아무것도 모르는 학부생을 대학원이라는 흥미로운 세계로 이끌어 주셨습니다. 교수님의 헌신적인 논문 지도와 아낌없는 연구 지원을 통해 제가 무사히 박사학위를 마칠 수 있었습니다. 그리고 박사학위 논문 심사위원으로 참여해주시고 소중한 조언을 주신 이보름 교수님, 신종원 교수님, 송영민 교수님께 감사드립니다. 또한, 바쁘신 와중에도 외국에서 자정 늦은 시간에 박사학위 심사에 참여해주시고 다양한 조언을 주신 김영래 (Young L. Kim) 교수님께 감사드립니다.

아낌없는 지지와 성원을 보내주신 가족 구성원분들께 감사의 말씀을 드립니다. 먼저, 조부모님께 감사드립니다. 초등학교 때부터 대학교까지 조부모님의 보살핌을 통해 건강히 자랄 수 있었습니다. 또한, 힘든 일 하시면서도 무한한 사랑과 관심을 주신 부모님께 감사드립니다. 부모님의 헌신 덕분에 부족함 없이 연구와 학업에 전념할 수 있었습니다. 그리고, 두 분의 누님들께도 감사의 마음을 전합니다. 누님들의 너그러움과 넓은 이해 덕분에 아무 다툼 없이 서로 도와가며 잘 지낼 수 있었습니다.

인포넷 연구실에서 동거동락 하며 오랜 시간을 함께 보낸 구성원 분들께도 감사의 마음을 전합니다. 먼저, 저의 영원한 사수 응비형께 감사 드립니다. Z 세대 최전선의 말 안 듣는 부 사수 데리고 가르치시느라 힘드셨을 것으로 생각합니다. 그럼에도 아낌없는 조언과 격려 주셔서 감사합니다. 그리고, 연구실의 다양한 일에 대하여 많은 경험과 지식으로 도움을 주신 승찬이형께 감사드립니다. 또한, 힘든 일 있을 때 불평불만을 격없이 받아주신 상준이형께 감사드립니다. 제가 졸업을 앞두고

이기적으로 행동했을 수도 있는데, 그럼에도 많은 도움과 정보를 주신 주성이형, 해웅이형, 재혁이형, Yaseen 형께 감사드립니다. 힘들 수 있는 연구실 생활을 뚝심 있게 해내고 있는 하영이 졸업까지 얼마 남지 않았으니 힘내라 전하고 싶습니다. 이외에도 Pavel 형, Wahidur 형, 형주, 성민이에게 감사의 마음을 전합니다. 마지막으로, 연구에 전념할 수 있게 행정적 일들에 많은 도움을 주신 정우정 선생님께 감사의 마음을 전합니다.

연구실에서 함께 생활했던 영학이형, Zafar 형, Asif 형, 승윤이형, 대영이형, 재원이형, 정이형, Rohit 형, 현준이형, 기원이형, 길준이형, 호현이형, 창윤이, 형성이, 그리고 정현준 교수님 모두 감사드립니다. 그리고, 연구 과제를 같이 수행하면서 광학 실험 경험을 쌓게 해주신 주건우 박사님, 이수경 박사님, 이용탁 교수님께 감사드립니다.

약 10 년 가까이 저와 기숙사 룸메이트를 하고 있는 병진이에게 고마움을 전합니다. 룸메이트 때문에 고생을 겪는 사람들도 많은데 병진이 덕분에 편안한 기숙사 생활을 할 수 있었습니다. 그리고, 광주과학기술원 학부부터 대학원 생활을 같이한 동기들, 병진이, 민승이, 철기, 연수, 국성이, 재훈이, 락희 그리고 석준이 모두 감사드립니다. 또한, 고등학교 동기이자 대학원 생활을 같이한 동주와 동현에게 감사의 마음을 전합니다.

마지막으로 사랑하는 지원이에게 감사의 마음을 전합니다. 박사과정 동안 옆에서 큰 힘이 되어 주었습니다. 항상 고맙습니다. 앞으로 더욱 행복하게 잘 지내자!

교수님, 가족, 애인, 동료분들의 도움과 헌신 덕분에 지금의 제가 있다고 생각합니다. 다시 한번 감사의 마음을 드리며 더욱 노력하고 발전해 나가겠습니다. 감사합니다.