Reconstructing Spectral Images from RGB-Images using a Convolutional Neural Network

Tarek Stiebel Simon Koppers Philipp Seltsam Dorit Merhof
Institute of Imaging & Computer Vision - RWTH Aachen University
tarek.stiebel@lfb.rwth-aachen.de

Abstract

Recovering high-dimensional spectral images taken with spectrally low-dimensional camera systems, in the extreme case RGB-images, has been of great interest for a variety of applications. An accurate spectral reconstruction is typically required to either achieve a better color accuracy or to improve object recognition/classification tasks. Almost all published work to date aims at performing a mapping from individual camera signals towards the corresponding spectrum. However, it might be beneficial to consider not only single pixels, but also contextual information.
Here, we propose a convolutional neural network architecture that learns a mapping from RGB- to spectral images. We trained the network on the largest hyper-spectral data set available to date [3] and analyzed the influence of different error metrics as loss functions. An objective evaluation of the performance in comparison to state of the art spectral reconstruction techniques is given by participating in the NTIRE 2018 challenge on spectral reconstruction [4].

1. Introduction

The task of recovering the spectral image from a low dimensional (e.g. RGB) spectral measurement has been of great interest for a variety of applications for a long time. In the 1990ies, several groups developed multi-spectral imaging [7, 8, 14]. One of the major questions at the time was how a multitude of images recorded with narrowband or even distinct broadband spectral filters can be used to recover the spectral reflection functions of individual pixels in the most accurate way.
Having knowledge about the reflection functions brings significant advantages, especially for color measurement and object recognition/classification. Regarding colorimetry, color in the context of human perception can not only be measured more accurately, but it also allows for an exact calculation of colors in conjunction with any light source. Thus, the dependency on the light source being present when the picture was taken is lifted. On the other hand, the additional spectral information may be used to identify previously indistinguishable objects.

Measuring the actual spectral object reflectances typically requires a controlled environment, even when using multi-spectral imaging. This is why most of the available data sets of spectral images, especially those containing outdoor images, do not provide the spectral object reflectances, but the combination of object reflectance and illumination. Separating such spectral images into spectral object reflectances and illumination is in general a complicated task and constitutes an active field of research [17].

The technique of multi-spectral imaging is not widely used despite its advantages. Major reasons are simply the cost and the fact that RGB cameras are typically considered to be sufficient for consumer end devices. Nevertheless, multi-spectral imaging systems are established in the professional sector, e.g. in the textile industry and in catalogue production. Multi-spectral line scanners are also available for professional applications which enable a high quality color image and/or object recognition, e.g. plastic classification in the context of recycling systems.
On the other hand, RGB-cameras are widely used and are most certainly capable of producing visually appealing images. Although it has been established for a long time that such imaging systems are not suited for the task of color measurement or spectral reconstruction in general, their performance is still of interest. For one because they are the devices which are actually available to anybody, but also since they are a baseline indicator multi-spectral imaging has to outperform.
The general consensus that RGB-cameras lead to significantly inferior spectral reconstructions is based upon mostly hand crafted algorithms or mathematical methods which describe a mapping from individual camera signals towards a spectrum in general. The most widely used methods are probably the Wiener deconvolution [8] and the application of learned basis functions [5, 13, 16]. It should be noted, that both methods have distinct requirements. While knowledge about the spectral sensitivity of the camera is essential

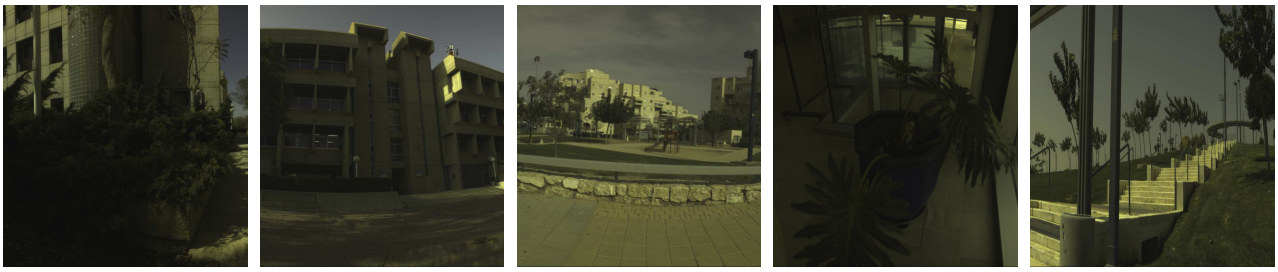


Figure 1: Examplary images from the ICVL dataset [3].

for the Wiener deconvolution, basis functions need to be actively learned from training data.

Within this work, machine learning is applied to describe the mapping from camera RGB-images to spectral images. Most importantly, a convolutional neural network architecture is proposed (CNN) that performs this mapping based not only on individual camera signals but also based on local contextual information. The proposed architecture is comparably simple, but convenient, robust to train and highly competitive.
The evaluation is performed on the largest hyper-spectral data base to date [3]. An analysis on the influence of three different metrics chosen as the loss function during training is provided. As a part of the NTIRE 2018 challenge on spectral reconstruction [4], the proposed CNN architecture was evaluated against state-of-the art competitors.

2. Spectral Reconstruction from RGB

Although several mathematical methods have been developed and tested in the context of multi-spectral imaging to perform spectral reconstruction, there was also a significant amount of research specifically targeting the task of spectral reconstruction from RGB-images. For example, Arad et al. [3] learn a dictionary based mapping which was improved by Aeschbacher et al. [1], Jia et al. [12] combine a non-linear dimensionality reduction of spectral data and a subsequent manifold mapping with machine learning to perform the actual reconstruction and Nguyen et al. [15] use a radial basis function network.

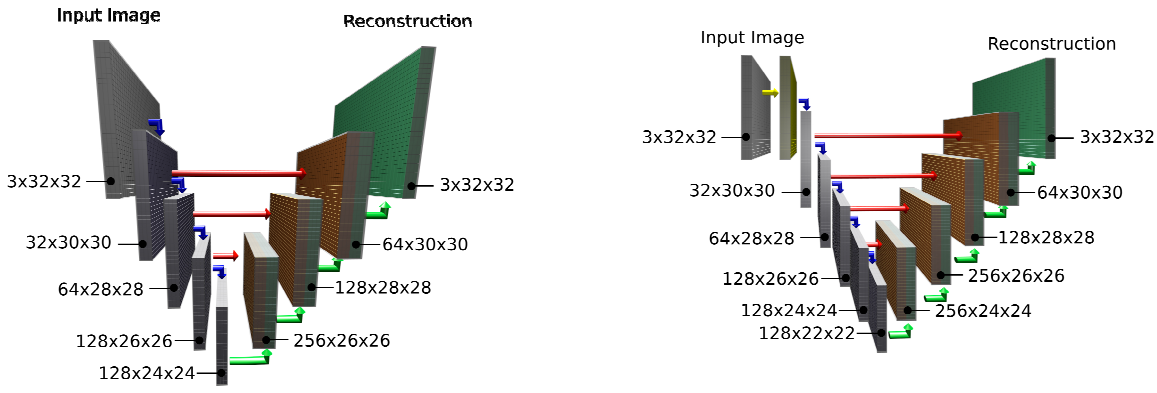
Still, all of these methods have in common that they perform a mapping based on a single RGB value. Instead, it might be beneficial to actually consider entire regions of RGB-values for a more sophisticated and stable mapping. Unfortunately, it is yet unclear how to actually integrate additional information from the neighboring pixels. Convolutional neural networks (CNNs) are a promising approach, however the lack of sufficiently sized data sets has prevented their application to spectral data in the past.
This has changed since the publication of the ICVL data set, which gave rise to first approaches based on CNNs. Galliani [6] uses a modified version of the Tiramisu architecture [11], which in turn is a modified Densenet [9]. Following the recent trends in machine learning, an approach using a generative adverserial network (GAN) has been proposed by Alvarez-Gila et al. [2].

2.1. Usage of Contextual Information

The idea of calculating the spectral reconstruction not only from individual RGB-values, but considering entire regions of pixels instead, is certainly not new. It is obviously advantageous in the presence of any kind of disturbances in real world images such as measurement noise, lens distortions, chromatic aberrations or even compression artifacts. However, there might also be contextual information available such as shading effects, which could increase the potential reconstruction quality. One might even think of a spectral reconstruction based on a previous object recognition step. For example, identifying a pixel as part of the sky could heavily limit the set of potentially corresponding spectra.

2.2. Network Structure and Design

We use a U-Net as a starting point [18]. The choice is motivated by the idea of combining the task of spectral reconstruction with semantic segmentation, a task the U-Net is known for. Although we do want to take advantage of contextual information, we believe that an actual two-step approach consisting of object recognition and subsequent spectral reconstruction is not yet possible. This is due to the limited amount of images within the provided data set, which is too small to robustly learn recognizing objects such as cars, buildings or trees to subsequently limit the choices for the actual spectral reconstruction. Instead, we are focusing solely on the very local neighborhood.
We currently consider spectral reconstruction to be mostly a regression task which benefits from classification. In contrast to pure classification tasks, the pooling layers within the standard U-Net will most likely lead to inferior results since information is actively thrown away. This is ac-



(a) Ideal Scenario

(b) Noisy Scenario

Figure 2: Visualization of used network architectures in case of RGB-images as input.

ceptable for classification tasks since the absolute numbers, which are computed in the end, are only of limited interest, whereas regression tasks actually have the goal of calculating the very precise absolute numbers. All pooling layers of the U-Net are therefore completely removed.
Following a similiar argumentation, there is no data normalization desired at any point.
One of the core ideas is to focus on local context information to enhance the spectral reconstruction results. This is further enforced by an input image size of 32x32 accepted by the network.
The resulting network structure is visualized in Figure 2a. The downsampling path of the network consists of convolutional layers having a kernel size of 3, a stride of 1 and no zero-padding, leading to each convolution reducing the input image size by 2 pixel. Each convolution is followed by a ReLU activation. The combination of these convolutions and a ReLU activation is visualized by blue arrows in Figure 2. The upward side of the network consists of corresponding transposed convolutions also having a kernel size of 3. Such transposed convolutions followed by a ReLU activation are drawn as green arrows. Skip connections are added everywhere but in the uppermost level and are visualized as red arrows.
The very first convolution that is applied takes the image as observed by a camera as input and outputs 32 channels. The subsequent two convolutional layers each double the channel count up to a final count of 128. Afterwards, the channel count remains constant until it is reduced again in the upward path, which is symmetric to the downward path. An initial hyperparameter search quickly revealed that an amount of five layers and a final filter count of 128 is ideal. There was no gain observed in increasing the filter count or the amount of layers any further.

The network has been implemented in PyTorch. The implementation provided within [10, 19] was initially used and modified.

2.3. Dealing with non-ideal Images

In a real world scenario, images can be expected to be noisy and subject to different disturbances. In order to become more robust against these disturbances, the network is slightly modified.
There are exactly two modifications made. First of all, a convolutional layer is added at the very start. It is meant to act as a simple pre-processing step. The output of this pre-processing is fed into the actual network and has as many channels as the input image. A kernel size of 5 has been found to be optimal (with a stride of 1 and a zero-pooling of 2). As a second modification, the original amount of five layers was increased by one to a new count of six. The final network structure is displayed in Figure 2b. The pre-processing step is represented by the yellow arrow.

3. Results and Discussion

In order to quantify the performance of the proposed architecture, it has been trained and evaluated on the largest hyper-spectral data set currently available.

3.1. Data

An extended version of the ICVL data set [3] is considered, as it was supplied during the NTIRE 2018 challenge on spectral reconstruction [4]. Next to the 203 hyper-spectral images, which are currently publicly accessible, there are also 53 newly collected images. The images are not used at their full spectral resolution corresponding to 519 channels, but in a downsampled version having 31 channels ranging from 400nm to 700nm in 10nm steps.

Some exemplary sRGB images are shown in Figure 1.
Since the dataset has been created with a rotating hyperspectral line scanner, the images appear slightly distorted. Corresponding camera images were computed for each spectral image using available spectral sensitivity functions of camera systems by using the formula

$$\chi_i = Sr_i \quad \forall\, i \in I, \tag{1}$$

with r_i being a 31-dimensional vector denoting the spectrum corresponding to the i'th pixel within the image I. The matrix S represents the spectral sensitivity and has q rows and 31 columns, where q denotes the spectral dimensionality of the camera system, e.g. $q = 3$ in case of a RGB-camera. The multiplication of sensitivity and spectrum for each pixel i results in the corresponding camera signal χ_i, a vector of dimension q.
It is important to stress that pixel interpolation is not considered. In reality, single chip cameras have to pay for their increased spectral resolution with spatial resolution. The RGGB Bayer pattern is the most prominent example which implies that information about a channel is not available at every pixel. In order to not be influenced by different filter array designs, we follow the same practice as in [2, 3] and compute all channels at every pixel.
Different RGB-cameras are considered. First of all, a camera of our own is used. The respective sensitivity function was measured using a monochromator setup. The resulting relative sensitivity function is displayed in Figure 3.

In addition to our own simulated image pairs, we also used the image pairs supplied during the NTIRE 2018 challenge on spectral reconstruction [4]. The challenge consisted of two tracks, which were called "Clean" and "Real World". While the RGB-images within the track Clean have been computed in the same ideal way we computed ours but with a different sensitivity, the RGB-images within the track Real World additionally contain noise and JPEG-compression artifacts. The data provided by the challenge offers the possibility to serve as a benchmark test, comparing our method against others.

3.2. Training Details

The entire spectral image set was split into three sub sets each consisting of approximately the same amount of images. For both cameras, as well as for both challenge tracks, a network as described in Section 2.3 was trained from scratch on each of the subsets and evaluated on the other two. The training was performed on a GTX 1080 TI and took roughly 3h for an individual network. The patch size used within the training process was 32, the batch size was 10. Each model was trained for 5 epochs using adam optimization and, subsequently, another 5 epochs using SGD with an initial Nesterov momentum of 0.9. All training images were split into patches in a deterministic way, such that neighboring patches are located next to each other.
A very common question in machine learning is the choice of an adequate loss function. For spectral reconstruction, choosing the loss function is difficult as there is no generally accepted quality measure. The measure is always task-dependent: For example, there is the well accepted color error metric $\Delta E2000$ if accurate color measurement is desired.

The same argumentation makes it hard to quantify the quality of a reconstruction in general. Therefore, there will be a total of 4 metrics provided to assess the quality of reconstruction: Goodness of fit coefficient (GFC), root mean square error (RMSE), relative root mean square error (RMSErel) and mean relative absolute error (MRAE).
The perceptive color error metric $\Delta E2000$ is explicitly excluded in our evaluation, since its valid computation is unclear given the provided data. It would require an identical illumination in all images, as well as a certain spectrum which is known to appear as white to the human observer. Both requirements are not given. It is possible to compute the $\Delta E2000$ metric based on some arbitrarily chosen white reference which is simply assumed to hold for the entire data set. However, the error metric will change, when another spectrum within the data set is chosen as white.

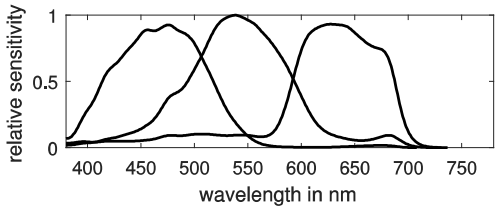


Figure 3: The relative spectral sensitivity function

3.3. Discussion

Table 1 displays the mean results for our RGB-camera for each fold. For all folds, the results are similar. This behavior is consistent for all evaluations we made.
Table 2 displays the reconstruction quality for all consid-

	RMSE	RMSErel	MRAE	GFC
Fold 1	15.776	0.02966	0.0153	0.9993
Fold 2	15.161	0.02733	0.0152	0.9995
Fold 3	15.071	0.02964	0.0151	0.9992

Table 1: Reconstruction quality for each fold when using our own camera.

Scenario	RMSE	RMSErel	MRAE	GFC
RGB Basler	15.336	0.0289	0.0152	0.9993
Challenge Track Clean	20.146	0.0382	0.01704	0.9988
Challenge Track World	27.557	0.05104	0.03081	0.9985

Table 2: The quality of reconstruction when using the network described in Section 2.2 trained on the respective image pairs using MRAE as loss function. The displayed metrics are the mean results over all folds.

loss	RMSE	RMSErel	MRAE	GFC
RMSE	15.0693	0.0263	0.0165	0.9996
RMSErel	15.1333	0.02755	0.0166	0.9995
MRAE	15.3366	0.0289	0.0152	0.9993

Table 3: The influence of the choice of different loss functions onto the reconstruction quality.

Network	RMSE	RMSErel	MRAE	GFC
Orig. Arch.	27.557	0.05104	0.03081	0.9985
Mod. Arch.	26.763	0.04892	0.03002	0.9987

Table 4: The quality of reconstruction can be improved in a real world scenario by using the slightly modified network described in Section 2.3.

ered scenarios, when MRAE was chosen as the loss during the training process. The values displayed represent the mean error metrics over all three folds. The results of our our RGB-camera are comparable to the images provided within the challenge. It is also notable, that the results in the track Real World of the challenge are worse than in the track Clean, which is expected.
In order to evaluate the influence of a different loss function on the training, the same training as before is performed with the important difference that each network is trained using RMSE or RMSErel as loss.
Table 3 displays the results for our camera averaged over all three folds. All in all, the results are comparable. Using the RMSE as loss function does lead towards better results when considering RMSE, RMSErel as well as GFC as an evaluation metric. There is no significant difference observable, when using RMSErel as loss instead of RMSE. On the other hand, using MRAE as a loss function leads to improved performance for the MRAE evaluation metric.

Next, the modified architecture described in Section 2.3 is compared against the original architecture of Section 2.2. Table 4 displays the averaged metrics over all folds for the respective networks, when considering the track Real World of the challenge. An improvement in all metrics could be achieved.

Finally, the proposed networks were used to participate in the NTIRE 2018 challenge on spectral reconstruction [4]. For each of the two tracks, a network was retrained on all available images using MRAE as loss function from scratch. There was no longer a data split performed beforehand. In case of the track Clean, the architecture described in Section 2.2 was employed, whereas for the track Real World, the architecture described in Section 2.3 was used. With regard to the track Clean, a final test score of MRAE= 0.0152 (RMSE= 16.191) was achieved. In the track Real World, a final test score of MRAE= 0.0335 (RMSE= 26.449) was achieved.

4. Conclusion

We have proposed a CNN architecture for the task of spectral reconstruction from RGB-images. It could be shown how the addition of a simple pre-processing layer enhances the quality of reconstruction in a real world scenario. Care has to be taken when choosing the loss function during the training process. The optimal choice is task-dependent. The proposed network architecture is comparably simple. Therefore, it is easy to implement and robust during the training process, while still showing a highly competitive performance. This could be validated by participating in the NITRE 2018 challenge on spectral reconstruction from RGB-images [4], achieving final scores corresponding to a top 4 ranking within both Tracks.

References

[1] J. Aeschbacher, J. Wu, and R. Timofte. In defense of shallow learned spectral reconstruction from rgb images. *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 471–479, 2017. 4322

[2] A. Alvarez-Gila, J. v. d. Weijer, and E. Garrote. Adversarial networks for spatial context-aware spectral image reconstruction from rgb. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 480–490, Oct 2017. 4322, 4324

[3] B. Arad and O. Ben-Shahar. Sparse recovery of hyperspectral signal from natural rgb images. In *European Conference*

on Computer Vision, pages 19–34. Springer, 2016. 4322, 4323, 4324

[4] B. Arad, O. Ben-Shahar, R. Timofte, L. Van Gool, L. Zhang, M.-H. Yang, et al. Ntire 2018 challenge on spectral reconstruction from rgb images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018. 4321, 4322, 4323, 4324, 4325

[5] V. Bochko, T. Jaaskelainen, and J. Parkkinen. Transform principal component analysis of spectral images. In *Conference on Colour in Graphics, Imaging, and Vision (CGIV)*, pages 120–124. Society for Imaging Science and Technology, 2004. 4321

[6] S. Galliani, C. Lanaras, D. Marmanis, E. Baltsavias, and K. Schindler. Learned spectral super-resolution. 2017. 4322

[7] J. Y. Hardeberg, F. J. M. Schmitt, and H. Brettel. Multispectral image capture using a tunable filter. In *Proc.SPIE*, volume 3963, pages 3963 – 3963 – 12, 1999. 4321

[8] B. Hill. Optimization of total multispectral imaging systems: best spectral match versus least observer metamerism. In *9th Congress of the International Colour Association*, volume 4421, pages 481–486, June 2002. 4321

[9] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, July 2017. 4322

[10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 2017. 4323

[11] S. Jégou, M. Drozdzal, D. Vázquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. 2016. 4322

[12] Y. Jia, Y. Zheng, L. Gu, A. Subpa-Asa, A. Lam, Y. Sato, and I. Sato. From rgb to spectrum for natural scenes via manifold-based mapping. In *International Conference on Computer Vision (ICCV)*, pages 4715–4723, Oct. 2017. 4322

[13] A. Kaarna, K. Tamura, S. Nakauchi, and J. Parkkinen. Non-negative bases for spectral color sets. In *2nd Int. Workshop on Image Quality and its Application*, pages 333–343, 2007. 4321

[14] Y. Miyake, Y. Yokoyama, N. Tsumura, H. Haneishi, K. Miyata, and J. Hayashi. Development of multiband color imaging systems for recordings of art paintings. In *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts*, 1999. 4321

[15] R. M. H. Nguyen, D. K. Prasad, and M. S. Brown. Training-based spectral reconstruction from a single rgb image. In *European Conference on Computer Vision (ECCV)*, pages 186–201. Springer, 2014. 4322

[16] W. Praefke and T. Keusen. Optimized basis functions for coding reflectance spectra minimizing the visual color difference. In *Third Color Imaging Conference*, pages 37–40. Society for Imaging Science and Technology, 1995. 4321

[17] A. Robles-Kelly. Single image spectral reconstruction for multimedia applications. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, pages 251–260, New York, NY, USA, 2015. ACM. 4321

[18] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 4322

[19] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 4323