

Wavelength-aware 2D Convolutions for Hyperspectral Imaging Supplementary Material

Leon Amadeus Varga, Martin Messmer, Nuri Benbarka, Andreas Zell
Cognitive Systems Group
University of Tuebingen
Tuebingen, Germany

leon.varga@uni-tuebingen.de, martin.messmer@uni-tuebingen.de,
nuri.benbarka@uni-tuebingen.de, andreas.zell@uni-tuebingen.de

1. Introduction

First, further details about the implementation of the method are given. In section 4, the bands of recordings of the cameras Specim FX 10 and Corning microHSI 410 Vis-NIR are compared. These two cameras were used in the experiments of application A. Then, to strengthen the argument of camera agnostic behavior, we provide a third set of experiments on the fruit set with two simulated cameras. Afterward, common objections are tackled. The supplementary material is rounded out by ablation study experiments that did not find a place in the main paper and a visualization of the prediction on the HRSS data set.

2. Method

In this section, we provide additional notes regarding the implementation. Further, we describe the number of learnable parameters for depthwise-separable convolutions in more detail.

3. Implementation of the Learnable Gaussian

For our implementation, we initialized the Gaussian distributions of the WROIs in a manner that the whole inspected wavelength range is covered. This is achieved by distributing the means $\mu_{t=0}$ evenly between the minimal (w_{min}) and the maximal inspected wavelengths (w_{max}). Further, the variance $\sigma_{t=0}^2$ is initialized with overlap:

$$\sigma_{t=0}^2 = \frac{1}{G}^2 \cdot (w_{max} - w_{min}) \quad (1)$$

Further, negative variances are prevented by using softplus [2].

3.1. Depthwise-Separable Convolution

Depthwise-separable convolutions[1] reduce the number of parameters by splitting up a convolution into a spatial-

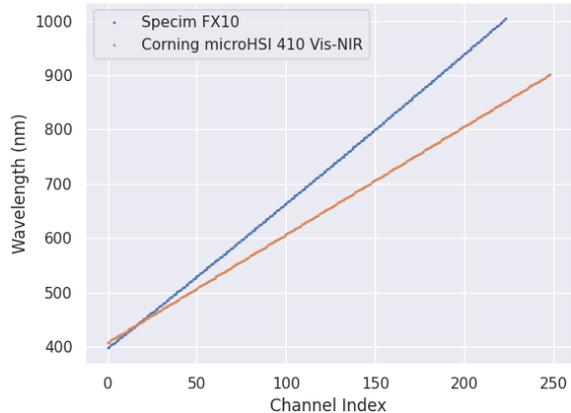


Figure 1: Assignment of wavelengths to channels for Specim FX10 and Corning microHSI 410 Vis-NIR

and a channel-based convolution, the overall relation between input channels and necessary weights still exists. Our baseline model DeepHS.net is based on depthwise-separable convolutions [1]. For a depthwise-separable convolution, the learnable parameters are necessary:

$$W_{ds} = (W_{spatial} \in \mathbb{R}^{C_{in} \times 1 \times K_x \times K_y}, \quad (2) \\ W_{depth} \in \mathbb{R}^{C_{in} \times C_{out} \times 1 \times 1})$$

4. Channels of Specim FX10 and Corning MicroHSI 410 Vis-NIR

For our approach, only the difference in the wavelength-channel-assignment is substantial (shown in Fig. 1). Other architectural decisions solve the handling of differences in spatial resolution.

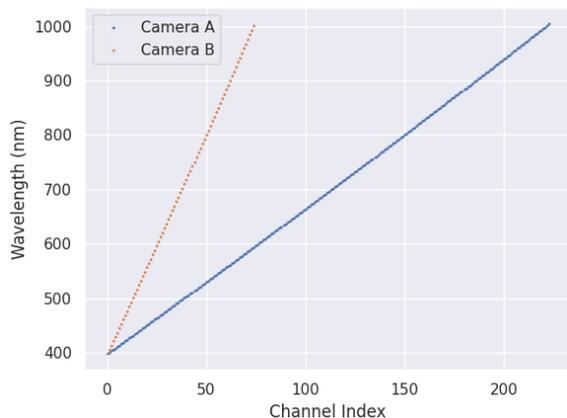


Figure 2: Assignment of wavelengths to channels for the synthetic data

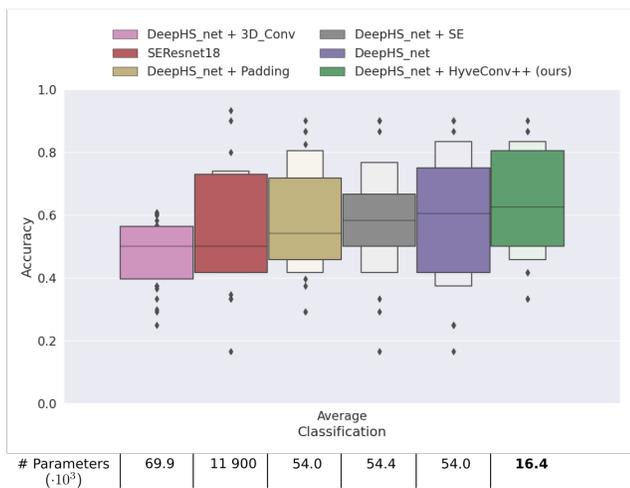


Figure 3: Overall accuracy on the ripening fruit data set with recordings generated for two synthetic cameras.

For the two selected cameras, the channels over index 50 differ widely between the two cameras and can result in invalid predictions without adequate handling. Also, the number of total channels slightly differs for the cameras. We show that the proposed method can handle both challenges.

5. Experiments for Camera-Agnostic Behavior on Synthetic Data

There is a lack of hyperspectral data sets with multiple camera recordings of the same scene. We evaluated our approach on a synthetic generated data set to support the claim of camera agnostic behavior. For this, a data set was created based on the Specim FX10 recordings of the ripening fruit

[3]. Two cameras are simulated using two channels’ subsets (shown in Fig. 2). We selected the channels of the subsets based on different step sizes for the channel indices. This mimics the real setup’s behavior, as shown in Fig. 1. The wavelengths of the latter channels differ more in contrast to the actual setup. Further, based on the sampling method of the channels of the two subsets, camera B contains fewer channels than Camera A. Therefore, this experiment setup’s challenge is slightly different from the configuration with two real cameras. We used the same models which were used in the main work.

Fig. 3 shows the final outcome of the experiments. Our method could again outperform the other approaches. For these experiments, the baseline model DeepHS_net with a linear interpolation could produce satisfying results, too. As already mentioned, the challenges of this experiment setup slightly differ from a real camera setup. But we could show that our model could handle both situations better than the other tested models.

6. Common Objections

Here we want to tackle the most common objections.

Additional hyperparameter The parameter G has a significant role in the method. But it is interpretable, and the default value was stable for two hyperspectral applications. We analyze the stability of the parameter in the ablation study (see section 7.2). Further, a basic understanding of the used wavelength range is necessary for hyperspectral application. Thus, a rough intuition of how many WROIs are required for a task is expectable. Further, the learned WROIs can be visualized, and it is possible to interpret them. For example, G can be increased or decreased based on their final overlap.

Slower inference The inference time is not affected by this modification. For the training, the prediction of the final kernel K is required in each step. For the inference, this can be done once per camera. No trainable part of the convolution is changing anymore. So the only difference in the inference time is the one-time prediction of the kernels for a new camera type.

Training stability The training was stable for two significantly different applications in our experiments. Also, the selection of the WROI worked reliably in these settings.

7. Ablation Study

This section analyzes the impact of a couple of design choices.

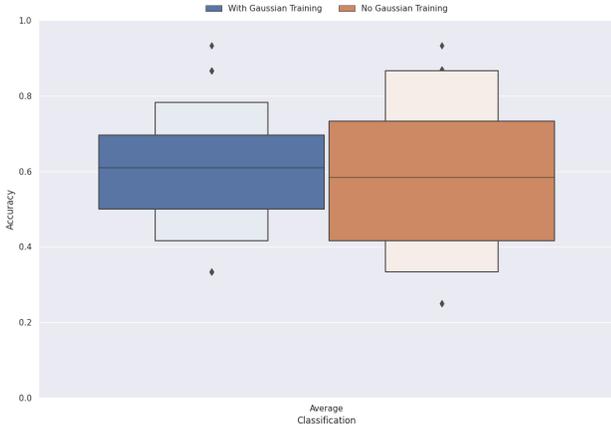


Figure 4: Performance with and without training of the Gaussian distributions on the ripening fruit data set.

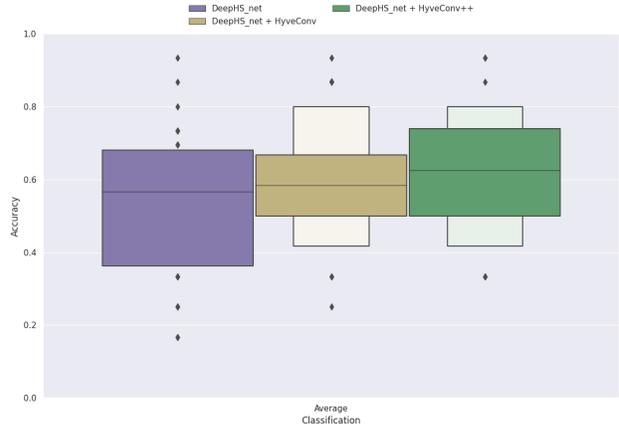


Figure 6: Performance of different approaches on the ripening fruit data set with recordings of the Specim FX10.

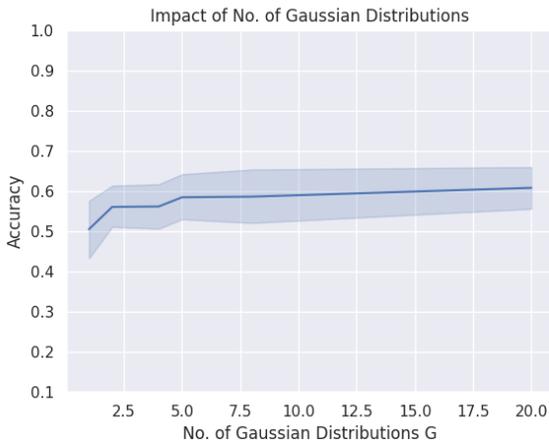


Figure 5: Performance of for different number of WROIs on the ripening fruit data set with recordings of the Specim FX10.

7.1. Training of the Gaussian Distributions

We evaluated whether training the Gaussian distributions is necessary for the fruit ripening data set. As a comparison, we used the distribution previously mentioned, which should cover the whole inspected wavelength range in an evenly distributed way. The result is visible in Fig. 4. There is a clear improvement in performance and stability.

7.2. Impact of Parameter G

Fig. 5 visualizes the impact of the parameter G , which defines the number of possible WROIs, on the performance of our model. This was tested on the ripening fruit data set and averaged over all setups (fruit type and classification type). A clear trend is visible. Increasing the num-

ber of Gaussian distributions increases the performance of the model. Two giant steps are visible: a Plateau between $G = 2$ and $G = 4$ and for $G \geq 5$. The increase is continuous for $G > 5$, but only slightly. Each jump in performance shows that the additional WROI provides helpful new information. Overall, the hyperparameter G seems stable above the threshold of 5. So, for a ripeness prediction for different fruit types, $G = 5$ is recommendable. For avocados, the classification seems more straightforward, and therefore already, $G = 4$ are sufficient (this was shown in section 5.1. of the main work). $G = 5$ also worked well for the second application B. As a result, $G = 5$ seems a reliable first choice, which the first training results can optimize.

7.3. Impact of the Method Extension

We checked whether the proposed extension of the basic method is useful. We compare the baseline model (DeepHS_net) with the basic method (HyveConv) and the extended method (HyveConv++). We did these experiments on the fruit ripening data set and averaged over all setups. The recordings of the Specim FX 10 were only used. Fig. 6 shows the performance of the different approaches. We tested the baseline model (DeepHS_net) with and without our camera agnostic convolution layer. Further, the impact of the proposed extension is shown. The continuous definition of the input channel space of the convolution (HyveConv) boosts the accuracy by around 2%. Allowing the convolution to share features through different output channels and the whole convolution layers (HyveConv++) boosts the model's performance by an additional 4%. Therefore, the extension seems reasonable.

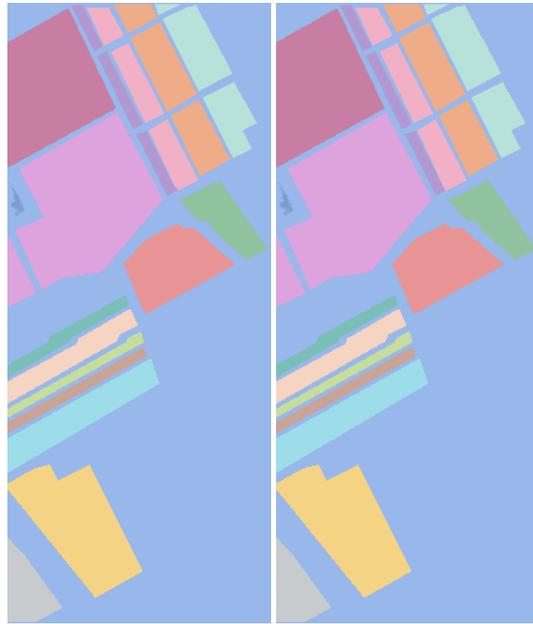
8. Visualization of the HRSS prediction result

For the HRSS data set, a visual comparison of the prediction and the ground truth is common, which can be found in Fig. 7 (Salinas), Fig. 9 (University of Pavia) and Fig. 11 (Indian pines). Especially for the Indian pines data set, some prediction errors of the model are visible. The corners of some areas are falsely classified.

Further, we provide for each model the training procedure of the camera filters and the final camera filters.

References

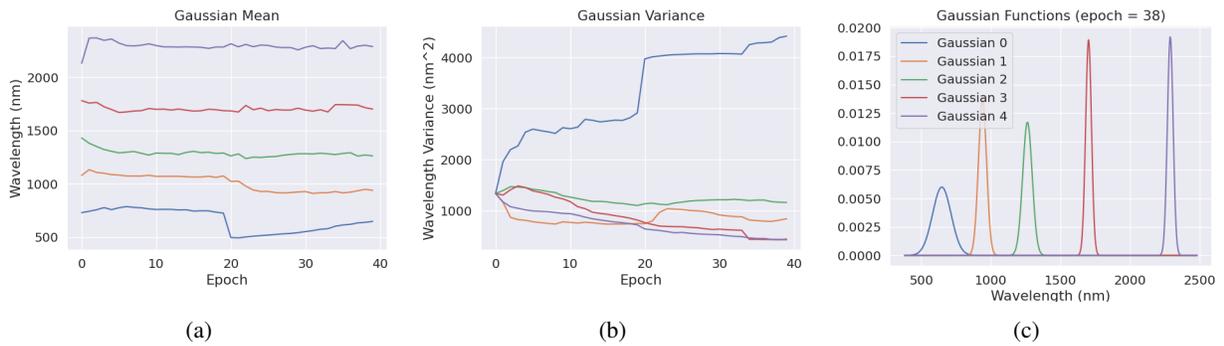
- [1] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1800–1807. IEEE Computer Society, 2017.
- [2] Kumar Shridhar, Felix Laumann, and Marcus Liwicki. A comprehensive guide to bayesian convolutional neural network with variational inference. *CoRR*, abs/1901.02731, 2019.
- [3] Leon Amadeus Varga, Jan Makowski, and Andreas Zell. Measuring the ripeness of fruit with hyperspectral imaging and deep learning. In *International Joint Conference on Neural Networks, IJCNN 2021, Shenzhen, China, July 18-22, 2021*, pages 1–8. IEEE, 2021.



(a) Prediction

(b) Ground Truth

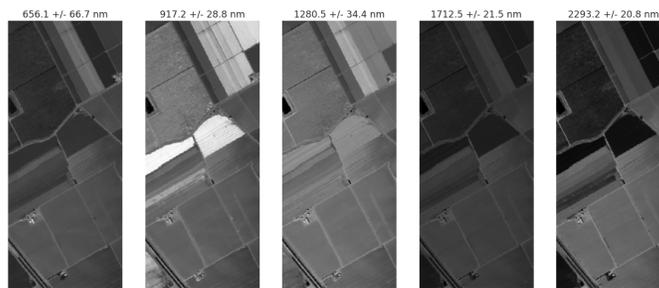
Figure 7: Prediction of DeepHS_net + HyveConv++ and ground truth for the segmentations mask of the Salinas data set.



(a)

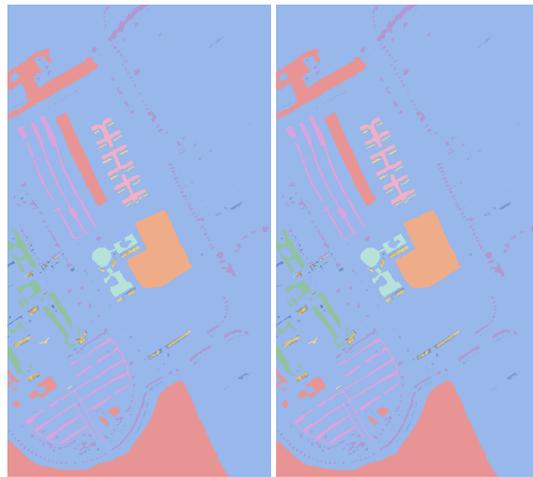
(b)

(c)



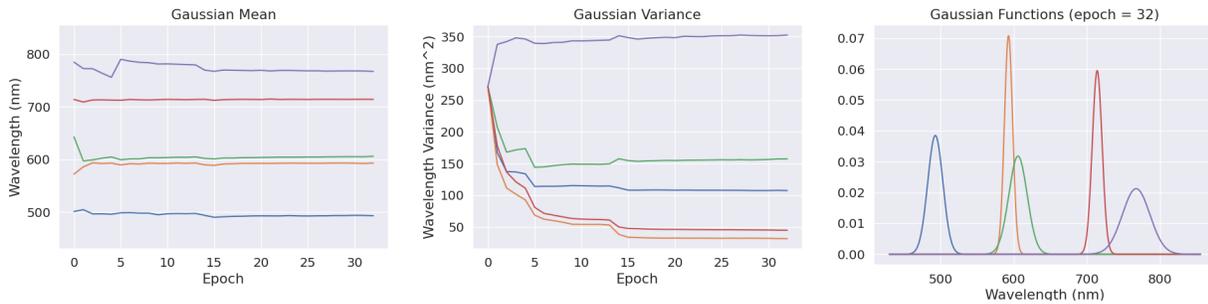
(d)

Figure 8: Training of the Gaussian distributions for the Salinas data set. (a) and (b) show the development of the mean and variance over training epochs. (c) shows the final Gaussian distributions, and these are applied as filters in (d).

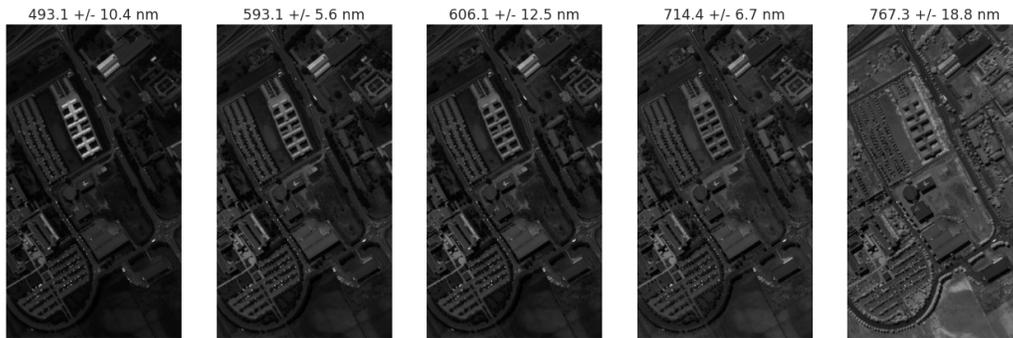


(a) Prediction (b) Ground Truth

Figure 9: Prediction of DeepHS_net + HyveConv++ and ground truth for the segmentations mask of the University of Pavia data set.



(a) (b) (c)



(d)

Figure 10: Training of the Gaussian distributions for the Pavia University data set. (a) and (b) show the development of the mean and variance over training epochs. (c) shows the final Gaussian distributions, and these are applied as filters in (d).

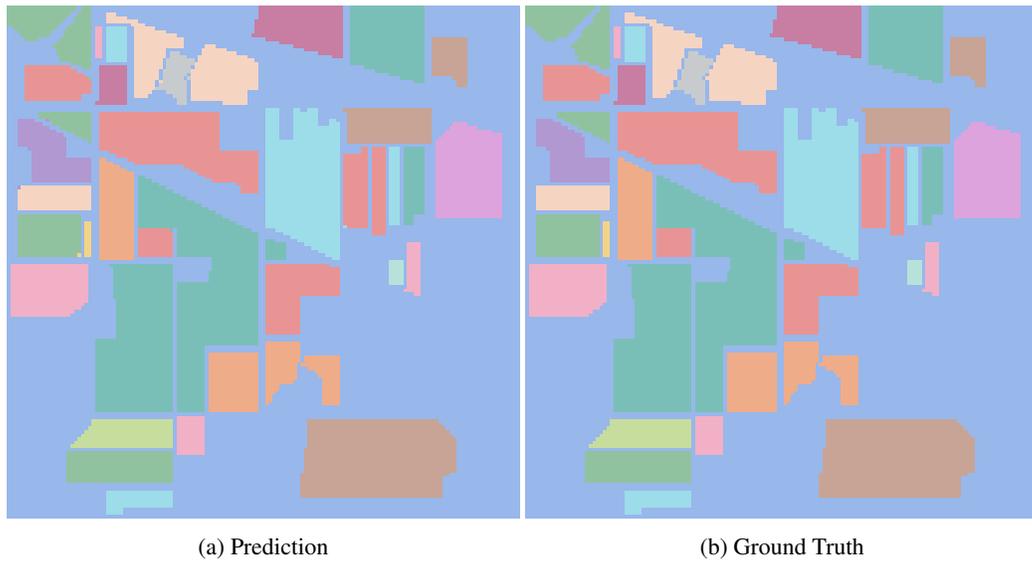


Figure 11: Prediction of DeepHS_net + HyveConv++ and ground truth for the segmentations mask of the Indian pines data set.

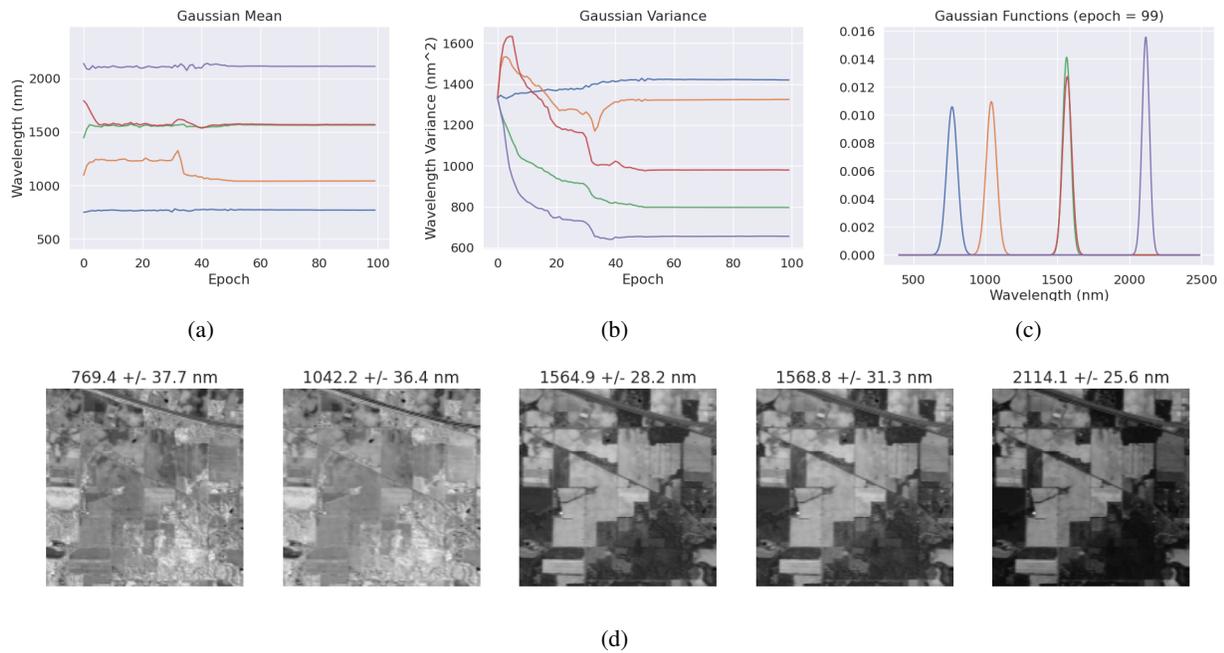


Figure 12: Training of the Gaussian distributions for the Indian pines data set. (a) and (b) show the development of the mean and variance over training epochs. (c) shows the final Gaussian distributions, and these are applied as filters in (d).