SEMANTIC SEGMENTATION OF NATURAL AND MAN-MADE FRUITS USING A SPATIAL-SPECTRAL TWO-BRANCHES-CNN FOR SPARSE DATA

Ulrike Pestel-Schiller, Ye Yang, Jörn Ostermann

Institut für Informationsverarbeitung, Leibniz Universität Hannover,
Appelstr. 9A, 30167 Hannover, Germany

ABSTRACT

We propose a CNN for semantic segmentation with classes which may be non-separable in the spatial domain, but distinguishable by additionally exploiting the spectral domain. In this spatial-spectral two-branches-CNN (SS2B-net), firstly, a spatial-branch-CNN exploiting the spatial domain and a spectral-branch-CNN exploiting the spectral domain are independently trained. Then, their parameters are fixed and their outputs serve as inputs for a new tiny CNN trained for classification into the given classes.

We reduce the number of hyperspectral bands from 186 to 96 by converting the VR-bands to RGB. Working on just RGB data in the spatial-branch-CNN, well-tried CNNs for RGB data can be applied. Finally, we use a pretrained VGG16-net to avoid overfitting caused by sparse data, together with a don't care class at the borders of classes to avoid overfitting caused by mixed pixels. In the spectral-branch-CNN, a small 1D CNN is designed and applied.

The segmentation results show improvements against the 3D hyper-U-net at class borders, at objects in their completeness, in mixed pixels caused by reflections and in wrongly classified objects. We support that by increasing the macro average recall from 0.90 to 0.97.

Index Terms— semantic segmentation, CNN, hyperspecral imagery

1. INTRODUCTION

Materials can be distinguished by their spectral characteristics of absorption or reflectance. Based on this, hyperspectral image (HSI) sensing allows the detection and identification of specific materials, e.g. for geological mapping, for monitoring agriculture and forest status, for environmental studies, for search and rescue services, for disaster managment or for surveillance. For each hyperspectral image pixel, a HSI sensor provides more than one hundred narrow and spectrally contiguous channels with a bandwidth of a few nanometers, ranging from the visible to infrared spectrum.

In an airborne context, for data transmission over a small data link, it is desirable to transfer just the classification result. That requires a well-working semantic segmentation already done on the platform followed by image compression e.g Joint Picture Expert Group (JPEG) [1]. We strive for a small data link.

In recent years, HSI has become an important application of machine learning [2] resulting in many different approaches for semantic segmentation to exploit the spectral information. In doing so, one challenge for all approaches is the huge number of spectral bands. It is common for HSI classification to apply one of many techniques to reduce the spectral dimensionality first [3].

For semantic segmentation of monochrome and RGB images, CNNs are known to be a powerful technique; in a sparse data context the number of learnable parameters has to be reduced, especially for applying CNNs in HSI.

Many approaches apply a Principal Component Analysis (PCA) across the spectral bands to reduce their number. In [4], a spectral-spatial, feature based classification is proposed where the spectral dimensionality is reduced with a specific technique called BLDE and the spatial features are extracted by first applying a PCA in spectral dimension over all bands and after that training a CNN on the first few principal component bands. Combining both, the resulting spectral-spatial features are finally classified together.

[5] compares different methods of using CNNs. A band reduction method is introduced by adding a selection layer for selecting the most interesting bands. On the remaining bands spatial features are trained and in parallel spectral features are trained pixelwise across all bands to result in a pixelwise classification. Furthermore, for semantic classification two U-nets [6], one for the bands of visible range (VR), the other for near-infrared range (NIR), are applied in parallel and concatenated and compared with a U-net applied on all bands (VNIR)), where all nets have a selection layer as first layer.

The aim of this work is to design a CNN for semantic segmentation under sparse data conditions. The classes are known and distinguishable in a spatial-spectral domain. Based on that, a CNN is proposed where first a network working in the spatial domain, called spatial-branch-CNN, and a network working in the spectral domain, called spectral-branch-CNN, are independently trained. Then, their parameters are fixed and their outputs serve as inputs for a new tiny union CNN trained for classification into the given classes.

We call the whole CNN a spatial-spectral two-branches-CNN (SS2B-net).

For many HSI applications sparse data is a challenge. Care has to be taken to avoid overfitting. It is essential to reduce the number of weights in a CNN. Our approach realizes this by totally separating the spatial-spectral domain in the training process. Furthermore, care has to be taken that mixed pixels covering multiple classes will not influence and disturb the classification result. To avoid that we introduce a dont'care area covering all mixed pixels at the border of classes.

We applied our proposed SS2B-net on HSI data with classes 'natural fruit', 'man-made fruit' and 'others' recorded with a HySpex VNIR-1800 at our institute. Thus, our method is applied to natural and man-made material intended to be visually indistinguishable. Nevertheless, the visible spectrum contains important information in the spatial domain while the detailed information of HSI in the spectral domain could irritate the CNN due to sparse data. Therefore, we reduce the number of input bands by converting the visible bands to an RGB-image. That allows us to use well-tried CNNs for RGB data to exploit the spatial domain.

The remainder of the paper is organized as follows: In Section 2 the method and the corresponding CNNs are introduced. Section 3 explains the recording and preprocessing of the data, the chosen data set for CNNs and the creation of label images. In Section 4 the evaluation criteria are defined. Section 5 shows the results.

2. METHOD

Fig. 1 shows the block diagram of the proposed system. The preprocessing comprises noise reduction and band reduction. Subsequently, the input data are trained in the spatial-branch-CNN and spectral-branch-CNN, respectively. The spatial-branch-CNN classifies into 'fruit', 'others' and 'dont'care', the spectral-branch-CNN into 'natural' and 'others'. These two outputs are then fed into the tiny union-CNN, which is trained to yield the desired semantic segmentation to classes 'natural fruit', 'artifical fruit', 'others' and 'don't care'.

2.1. Preprocessing of Data

After conversion of the HSI from radiance to 32-bit-float reflectance data, its amplitude depth is reduced by amplitude quantization to 8-bit-int which leads to a noise reduction. A further noise reduction by spatial averaging is done in the first layer of the spectral-branch-CNN. That allows us to have the same input data for both branches.

Semantic segmentation in the spatial-branch-CNN works well with RGB data. Thus, the VR-bands are converted to RGB. The spectral-branch CNN needs at most RGB and the VNIR area [7]. RGB allows to eliminate non-fruit colors, but avoids overfitting in the VR domain. RGB together with the bands 93 to 185 constitute the input data of the SS2B-net with a patch size of $224 \times 224 \times 96$. Hence, we have a band reduction from 186 to 96 spectral bands.

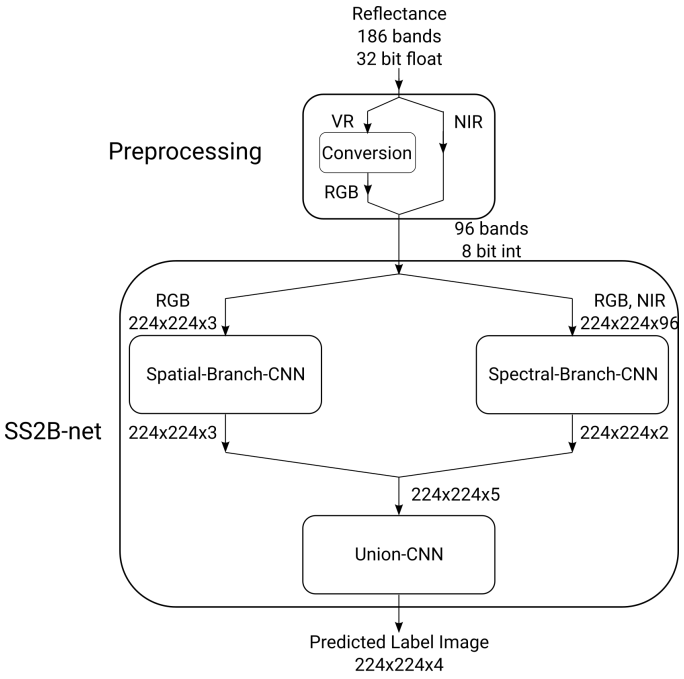

Fig. 1: Block diagram of preprocessing and SS2B-net

2.2. Spatial-spectral Two-Branches-Network (SS2B-net)

To master the challenge of sparse data and the huge 3D HSI data cube, we exploit the orthogonality of spatial and spectral information by training two networks separably, one network learning in the spatial domain and the other in the spectral domain. In this work, every convolution layer follows a non-linear activation.

With sparse data, the well known U-net [6] is well suited for semantic segmentation. Thus, a CNN based on a 2D U-net is chosen for the spatial domain. For the spectral domain, a small 1D CNN is designed and applied.

Each branch-CNN is trained on its own. Then the weights of the trained branch-CNNs are fixed and the outputs of the branch-CNNs are the input of the union-CNN as shown in Fig. 1 .

2.2.1. Spatial-Branch-CNN

The spatial-branch-CNN is a U-net based CNN, trained to classify into 'fruit', 'others' and 'don't care'. The stage of its encoder-decoder is 5. We train on RGB-data as converted from VR bands. The segmentation result of the original 2D-U-net [6] is further improved by the following steps, which accelerate the training and reduce the overfitting:

- Additional batch-normalization layers between the convolutional layer and the ReLU layer
- Additional class "don't care" at the border of classes which represents the mixed pixels to avoid irritations in the training process
- Replacing the U-net encoder by the pretrained VGG16 network [8],[9].

The output of the spatial-branch-CNN has a size of $224 \times 224 \times 3$.

2.2.2. Spectral-Branch-CNN

The spectral bands contain the information about natural objects and others.

The proposed spectral-branch-CNN is a 1D-CNN in spectral direction, trained to classify into 'natural objects' and 'others'. It consists of 3 convolutional layers with a filter size of 1x1x3 each. The output of the spectral-branch-CNN has a size of $224 \times 224 \times 2$.

2.3. Union-CNN

For achieving the intended semantic segmentation of natural fruits and man-made fruits, both classification results can be combined by concatenating the output layers resulting in a size of $224 \times 224 \times 5$. The joint classification is realized by a convolutional layer resulting in the final classification into the four classes 'natural fruit', 'man-made fruit', 'don't care' and 'others'. Finally, this last stage of the SS2B-net is trained while fixing the weights of the spatial-branch-CNN and spectral-branch-CNN. The output of the SS2B-net has a size of $224 \times 224 \times 4$.

2.4. Reference Network

Under sparse data conditions, segmentation of biomedical images often uses the U-net [6]. Since medical input images normally have only one component, the U-net uses 2D convolutional layers with a 3x3 convolutional kernel. For additionally learning spectral features from the hyperspectral imagery, we replaced all 2D-convolutional layers by 3D convolutional layers of size 3x3x3. The obtained 3D hyper-U-net with 5 stages is our reference-CNN.

2.5. Training options

All CNNs are trained with the same options.

The input size of the CNN is 224x224x96. We use the *Adam* (adaptive moment estimation) optimizer with an initial learn rate of 0.0001.

We train on 7168 (7 images $\times$ 1024) HSI samples extracted from the seven training images. They were randomly split into training (80%) and validation set (20%). 1024 per image turned out to be the best compromize between effective training and low training time.

The relative class frequencies are given with 81.5% for 'others', 17.2% for 'fruit' and 1.3% for 'don't care'. To compensate for the imbalanced class frequencies the classes are weighted with the inverse frequencies. We implemented the software in MATLAB. The SS2B-net will be available at our homepage.

3. DATA

Our dataset consists of ten images depicting natural and man-made fruits together with other artificial objects like a cup, a box, a socket or man-made flowers. All HSI data of this work was recorded with the hyperspectral sensor HySpex VNIR-1800. It will be available at our homepage.

3.1. Data Set for CNNs

After preprocessing, the final data set contains the 10 hyperspectral images with corresponding label images, split into 7 of size 2072x1800x96 for training and the other 3 of size 1199x1800x96 for testing.

To increase the dataset, data augmentation using random cropping, random rotation and random flipping is used in the training process. For testing, interesting parts of the test images were cropped, resulting in the test set. Fig. 2, line (a) shows examples.

3.2. Label Images

For semantic segmentation, we have to generate label images of our HSI data. We use a semi-automatic labelling method to achieve the required label images.

For their creation we start with the RGB image of the HSI by applying robust matting [10] to obtain a binary image.

For the spatial-branch-CNN, we first get binary label images with classes 'fruit' and 'others'. To avoid irritations in the training process caused by mixed pixels at borders, we add a third class 'don't care' at the borders of classes by using the trimap algorithm [11].

For the spectral-branch-CNN, we generate binary label images with classes 'natural fruit' and 'others'.

For the union-CNN, we create label images with classes 'natural fruit', 'man-made fruit', 'others' and 'don't care' from the label images of the spatial-branch-CNN and the spectral-branch-CNN.

For the 3D-hyper-U-net, we generate label images with classes 'natural fruit', 'man-made fruit' and 'others' by combining the above binary label images.

4. EVALUATION

The test set is evaluated with the following evaluation criteria.

4.1. Visualization of label images

To detect specific segmentation errors each output image of the CNNs called predicted label image is visually compared with the corresponding label image, acting as ground truth.

4.2. Intersection over Union

In semantic segmentation, the well-known *Intersection over Union* (IoU) of class i is defined as the intersection between the predicted pixels of class i and the labelled pixels of class i compared to their union as

$$IoU_i = \frac{TP_i}{TP_i + FP_i + FN_i} \tag{1}$$

where TP_i means true positive, FP_i false positive and FN_i false negative each relating to class i.

Based on Eq. 1, we define the *mean Intersection over Union* $mIoU$ by averaging IoU_i over all N classes as

$$mIoU = \frac{1}{N} \sum_{i=1}^{N} IoU_i. \tag{2}$$

4.3. Macro Average Recall

The *Recall* R_i *of class* i is defined as correctly predicted pixels of class i over labelled pixels of class i i.e.

$$R_i = \frac{TP_i}{TP_i + FN_i}. \tag{3}$$

The *Macro Average Recall* (MAR) [12] is calculated by averaging R_i over all N classes

$$MAR = \frac{1}{N} \sum_{i=1}^{N} R_i \tag{4}$$

Because of the imbalanced class frequencies, we use the MAR instead of accuracy.

4.4. Normalized Confusion Matrix

The *Normalized Confusion Matrix* as an extension of the Recall of class i is a table with rows representing label image classes and columns representing predicted label image classes. The main diagonal contains the *Recalls* R_i. The other positions show the amount of false prediction to a specific class together with the additional information which class is predicted instead. This information allows a detailed analysis of the segmentation result.

5. RESULTS

First we compare SS2B-net with 3D hyper-U-net. After that we analyze the SS2B-net in more detail.

In Fig. 2 selected results of SS2B-net and 3D hyper-U-Net are visualized. Row (a) shows the RGB-form of HSI, row (b) the label image of trimap, row (c) the segmentation result of 3D hyper-U-Net and row (d) the segmentation result of SS2B-net. Dark blue indicates 'others', light blue 'natural fruit', green 'artificial fruit' and yellow 'dont care'. Columns 1 and 2 contain 'others' and columns 3 to 6 fruits.

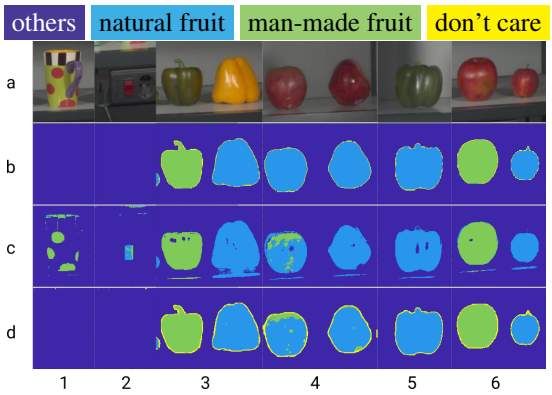

Fig. 2: Examples of image results in columns 1-6
(a) RGB (b) Label image (c) 3D hyper-U-net (d) SS2B-net

As Fig. 2 shows, using SS2B-net removes false segementation of

- reflections of fruits on the shelf (cols. 3-6)
- overfitting to false 'fruit' (cols. 1-2)
- overexposure points to background (cols. 3, 6)

and improves the segmentation of

- fruits in their completeness (col. 4)
- borders of all fruits in their smoothness

It should be mentioned that some false-looking segmentation areas belong to a sticker (Fig. 2, col. 4: apple on right side) and to dried fruit parts (Fig. 2, col. 4: pomegranate on left side)

Table 1 shows the IoUs of the SS2B-net. The IoU of class "don't care" is much smaller than the other classes caused by the width of the border which is two pixels in the label image and three pixels in the segmentation result. That reduces the mIoU to 0.836, but does not damage the fruit segmentation.

Table 1: IoU of SS2B-net and 3D hyper-U-net

Net	mIoU	Others	Natural Fruit	Man-made	Don't care
3D hyper-U	0.821	0.949	0.799	0.716	-
SS2B	0.836	0.992	0.940	0.923	0.490

Table 2 shows the Normalized Confusion Matrix of the SS2B-net, which shows how many pixels of each class were correctly segmented and to what percentage pixels were wrongly segmented as other classes. The Macro Average Recall of the SS2B-net is 0.970. Table 2 confirms the above results by showing the highest false segmentations from 'natural fruit' to 'man-made fruit' caused by the sticker and mummification and from 'fruits' to 'dont'care' caused by the unequal pixel width of the border in label and predicted segmented image.

Table 2: Normalized Confusion Matrix of SS2B-net

	Others	**Natural Fruit**	**Man made**	**Don't care**
Others	**0.992**	0.000	0.000	0.008
Nat. fruit	0.000	**0.952**	0.025	0.023
Man-made	0.000	0.019	**0.957**	0.024
Don't care	0.010	0.005	0.007	**0.978**

Table 3 shows the Normalized Confusion Matrix of the 3D hyper-U-net. The Macro Average Recall of the 3D hyper-U-net is 0.901.

Table 3: Normalized Confusion Matrix of 3D hyper-U-net

	Others	**Natural Fruit**	**Man-made Fruit**
Others	**0.973**	0.014	0.013
Natural fruit	0.080	**0.901**	0.019
Man-made Fruit	0.129	0.042	**0.829**

By comparing Table 2 with Table 3 we see the improvement of the accuracies of all classes by using SS2B-net and that especially the false segmentation of 'fruits' to 'others' is improved. Further improvement could be achieved by a postprocessing exploiting the spatial information obtained by enclosed 'don't care' areas.

6. CONCLUSION

Against overfitting caused by sparse data, we reduce the number of weights in the CNN by proposing a spatial-spectral two-branches CNN where the spatial and spectral domains are separated into a spatial-branch-CNN and a spectral-branch-CNN. For further preventing overfitting, we reduce the number of spectral bands at the input of the SS2B-net and 3D hyper-U-net from 186 to 96 by converting the VR bands of HSI to RGB.

Working on just RGB data in the 2D U-net based spatial-branch-CNN, we use a pretrained VGG16-net to avoid overfitting caused by sparse data, together with a 'don't care' class at the borders of classes to avoid overfitting caused by mixed pixels.

The segmentation results of the predicted label images show improvements at the borders of fruits, at fruits in their completeness, in mixed pixels caused by reflections and in wrongly classified objects.The Macro Average Recall is increased from 0.90 to 0.97. In future work, generally, a stronger more adapted band selection in the spectral-branch CNN is possible and should be applied.

7. REFERENCES

[1] W. B. Pennebaker, *JPEG still image data compression standard*, Van Nostrand Reinhold, New-York, 1993.

[2] N. Audebert et al., "Deep learning for classification of HS data: A comparative review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 159–173, 2019.

[3] W. Sun et al., "Hyperspectral band selection: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, pp. 118–139, 06 2019.

[4] W. Zhao et al., "Spectralspatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE TGRS*, vol. 54, no. 8, pp. 4544–4554, 2016.

[5] S. Trajanovski et al., "Tongue tumor detection in hyperspectral images using deep learning semantic segmentation," *IEEE T-BME*, vol. 68, no. 4, pp. 1330–1340, 2021.

[6] O. Ronneberger et al., "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015.

[7] U. Pestel-Schiller et al., "Determination of relevant hyperspectral bands using a spectrally constrained CNN," in *11th WHISPERS, Paper 15*, Mar. 2021.

[8] Deng et al., "Imagenet: A large-scale hierarchical image database," in *2009 IEEE CVPR*, 2009, pp. 248–255.

[9] K. Simonyan et al., "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.

[10] J. Wang et al., "Optimized color sampling for robust matting," in *2007 IEEE CVPR*, 2007, pp. 1–8.

[11] C.-L. Hsieh et al., "Automatic trimap generation for digital image matting," in *2013 APSIPA ASC*, 2013, pp. 1–5.

[12] Margherita Grandini, Enrico Bagli, and Giorgio Visani, "Metrics for multi-class classification: an overview," 2020.