Hyperspectral Image Reconstruction Employing TV-loss and Smooth L1-loss in SSR-NET

Author:
Maria Jose Rueda Montes

Supervisor:
Hao Wang
Guoxia Xu

***Abstract*—Hyperspectral image reconstruction is a technique based on the fusion of high resolution multispectral images (HR-MSI) and low resolution hyperspectral images (LR-HSI) with the aim of obtaining high resolution hyperspectral images (HR-HSI) at the output. This method is very popular since hyperspectral image capture sensors sacrifice the increase in the capture of spectral bands in exchange for resolution, generating undesirable noise in the final images. Although several fusion techniques have emerged to achieve HR-HSI, new methods based on deep learning, especially in convolutional neural networks (CNN), are obtaining pioneering results for the reconstruction of hyperspectral images. This article studies the implementation of two denoising algorithms, Total Variation (TV) loss function and Smooth L1-loss, as a proposal to improve the quality of the final fused image. To prove the effectiveness of our implementations, three different networks have been created, using SSR-NET as a reference in which they have been implemented: 1) TV-loss loss; 2) Smooth L1-loss; 3) TV-loss plus Smooth L1-loss. After comparing the results with respect to the original SSR-NET, in five different HSI databases, Botswana, Urban, Pavia Center, Pavia University, and Kennedy Space Center, it has been verified that the proposed implementations represent an improvement in the results, being TV-loss more effective for databases with high spatial resolution per pixel, and TV-loss plus Smooth L1-loss with better results for databases with lower resolution.**

***Index Terms*—Convolutional neural network (CNN), hyperspectral image (HSI), multispectral image (MSI), image reconstruction, total variation (TV) loss function, Smooth L1-loss function.**

1 Introduction

Hyperspectral images (HSI) are characterized by the hundreds of spectral bands obtained from a scene. The high spectral resolution of this type of images, allows good results in a large number of applications suchlike remote sensing and computer vision tasks [1]. The reason behind successful results of hyperspectral imaging is related to its accurate identification of materials since each material has a different reflectance at each wavelength, capture images at high spectral resolution and wide spectral range is an advance discriminating between different materials in a scene [2]. However, due to the limitations of imaging sensors, long exposures capturing hyperspectral images are necessary, causing a spatial resolution deficiency due to the signal-to-noise-ratio (SNR) generated, which leads to the low spatial resolution of hyperspectral images (LR-HSI). On the other hand, imaging sensors can obtain an image with a higher spatial resolution but a worse spectral resolution, such as multispectral images (MSI). As a solution to improve the quality of hyperspectral images, the concept of hyperspectral and multispectral fusion emerged. In this way, the high spatial resolution of multispectral images (MSI) (HR-MSI) to reconstruct high-spatial-resolution HSI (HR-HSI) is used. There are several approaches to HSI and MSI fusion, including pan-sharpening-based methods, matrix factorization-based methods and tensor-based methods [2]. However, compared to these traditional methods, new research based on deep learning using convolutional neural networks (CNN) are achieving excellent results improving the quality and performance of the final fused image.

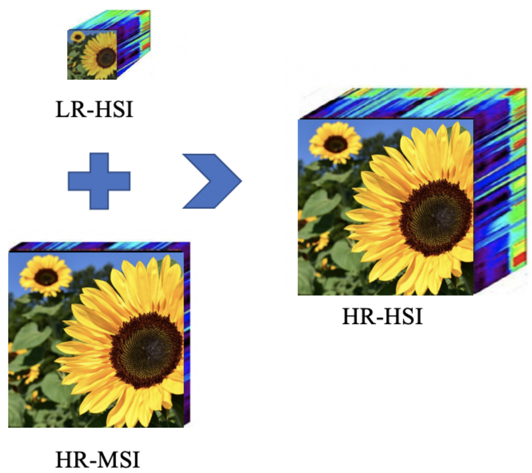

Figure 1: LR-HSI and HR-MSI fusion for obtaining HR-HSI.

The CNN models for LR-HSI and HR-MSI fusion can be divided into input-level fusion model and feature level fusion model. In feature level fusion approaches, first the spatial features are extracted from HR-MSI and the spectral features from LR-HSI. Both features are fused in order to reconstruct the HR-HSI. In [3], Shao *et al.* proposed a model with two branches network to extract the HR-MSI and LR-HSI information separately. In [4], Yang *et al.*, use two branches for extracting the spectral features of each pixel in LR-HSI, and its correspondent spatial neighborhood in HR-MSI, to take advantage of the spatial correlation. Then they fuse the information in a fully connected layer. In [5], Han *et al.* use a multi-scale CNN system, which reduce the HR images and increase the feature sizes of LR in a gradually way. On the other hand, for input-level fusion models, the LR-HSI and HR-MSI images are fused before passing through the neural network for obtaining the final fused HR-HSI. Normally, before the fusing, interpolation is applied to the LR-HSI to generate the same size as HR-MSI. Works of Masi *et al.* [6], or Palsson *et al.* [7], use the input-level fusion, where the preliminary fused LR-HSI and HR-MSI are used as input for a super-resolution CNN, SRCNN, and PCA prior for reducing the dimensions of the fusion. In [8], Dian *et al.* present a DHSIS method to reconstruct HR-HSI, where they map the first fused HR-HSI to the reference HR-HSI using the priors learned from a deep CNN-based residual learning to regularize the fusion. SSR-NET, created by Zhang *et al.* [9], incorporates three different modules: Cross-Mode Message Inserting (CMMI), Spatial Reconstruction Network (SpatRN), and Spectral Reconstruction Network (SpecRN). The main objective of the CMMI module is to produce a preliminary concatenated image that combines the spatial and spectral information of MSI and HSI, the SpatRN module allows the reconstruction of spatial information with the use of spatial edge loss ($\mathcal{L}_{spat}$), and finally the SpecRN module pays attention to the reconstruction of spectral information with spectral edge loss ($\mathcal{L}_{spec}$). This network achieve superior results than seven state-of-the-art methods, which are CNMF, LTTR, MSDCNN, TFNet, ResTFNet, SSFCNN and ConSSFCNN. Due to the good results obtained by this network, and the ease of executing it and observing the results of the changes quickly, we have choose SSR-NET as reference in this paper to improve the quality of the results in obtaining HSIs-HR.

Otherwise, since the bad resolution after the capture of HSIs is mainly influenced by the noise that is generated in the different bands, it is logical to pay attention to the implementation of methods that solve this problem. Total Variation (TV) regularization is one important technique used for denoising in HSIs, this method of regularization obtain very good results in image processing, being an important algorithm that achieves denoise since exploits the spatial correlation in each band and it works on the preservation of edge information and spatial smoothness [10]. In [10], He *et al.* use total variation regularization for the hyperspectral image restoration, where they propose a mixed-noise removal method that integrates TV regularization combined with the low-rank matrix factorization model. They use the low-rank model to study spectral correlations, while TV regularization uses the piecewise smooth of HSI spatial information. In [11], Aggarwal *et al.* introduce spatio-spectral total variation to develop an algorithm that reduces mixed noise. In this case they use total variation for the spatial dimension, along the height and width, and total variation for the spectral information. To improve video super resolution (VSR), [12] Chadha *et al.* introduce a generative adversarial network (GANs) with a space-time approach, incorporating a four-fold (MSE, perceptual, adversarial and TV) loss function to capture the fine details of the image, choosing TV-loss as denoising loss function. Due to the importance of TV loss in reducing noise in HSI images, and since it has been used successfully for this purpose in previous papers, we have integrated TV-loss to SSR-NET.

Since the quality of the final image depends on the choice of loss function, in this paper we have also studied the incorporation of Smooth L1-Loss as a substitute for MSE-loss. MSE-loss function is one of the most used methods, the algorithm of which helps to improve the PNSR metric, however, this metric is not representative of the spatial features of the image [12]. Smooth L1-loss also called Huber loss, is less sensitive to outliers than MSE-loss, generating good results in reducing noise in images. In [13], Chlewicki *et al.* proposed the Smooth L1-loss applied to image reconstruction, where they achieved the reduction of the noise without greatly affecting the contrast of the image. This is due to the fact that Sooth L1-loss penalizes the differences between the values of the neighboring pixels, without penalizing large differences between the values, which correspond to the edges of the image [13].

In summary in this paper are proposed the following contributions:

1) We incorporates TV-loss function for the first time in the CNN SSR-NET to improve the quality of the generated HSI.
2) On the other hand, the substitution of MSE-loss function for Smooth L1-loss will also be studied.
3) Compared with the results of SSR-NET, an improvement in the results is observed after applying TV-loss and Smooth L1-loss, in Botswana, Urban, Pavia Center, Pavia University and Kennedy Space Center databases.

This article is organized as follows: Section 2 explains the methodology used, summarizing how SSR-NET works and describing the built-in algorithms, TV-loss and Smooth L1-loss. In Section 3 the experimental results are explained in five dates. And finally in Section 4 the conclusions are shown.

2 Methodology

In this section we will start by summarising the explanation of the different components of the SSR-Net in order to understand how they influence the new components introduced in this project. This network consists of three differentiated modules: Cross-Mode Message Inserting, Spatial Reconstruction Network with Spatial Edge Loss and Spectral Reconstruction Network with Spectral Edge Loss. On the other hand, the implementation of the Total Variation Loss (TV-loss) and Smooth L1-loss algorithms is also explained in this Section.

A. Cross-Mode Message Inserting

The objective of this module is to process the HR-MSI and LR-HSI images that are going to serve as input to the network, since in this case the network is of the input-level fusion type. For this purpose HR-MSI is unsampled until reaching the size of LR-HSI and then merging both images using bilinear interpolation. In this way a hypermultiple spectral image (HMSI) is obtained, $\mathbf{Z}_{pre} \in \mathbb{R}^{H \times W \times L}$. Thus, this new set of images has the spatial and spectral information of LR-HSI and HR-MSI. Then a 3×3 and with 1 stride convolutional layer is applied to the images in order to combine spectral and spatial information.

B. Spatial Module

In this module, based on the reconstruction of spatial information, a new convolutional layer, SpatRN, with 3×3 and 1 stride characteristics is applied to $\mathbf{Z}_{pre}$, obtaining $\mathbf{Z}_{spat}$. This layer generate the edge maps of HMSI, utilizing $\mathcal{L}_{spat}$ which use the spectral information to update the weights. $\mathcal{L}_{spat}$ operates as follows

$$\mathbf{E}_{spat\_height}(i, j, k) = \mathbf{Z}_{spat}(i+1, j, k) - \mathbf{Z}_{spat}(i, j, k), \tag{1}$$

$$\mathbf{E}_{spat\_width}(i, j+1, k) = \mathbf{Z}_{spat}(i, j, k) - \mathbf{Z}_{spat}(i, j, k), \tag{2}$$

where i, j and k correspond to the dimensions of the data set, being the height, width and spectral bands, respectively. For this first pair of equations, in Equation 1 it is calculate the edge map in the height dimension (i), $\mathbf{E}_{spat\_height} \in \mathbb{R}^{(H-1)\times W\times L}$, while Equation 2 obtains he edge map in the width dimension (j), $\mathbf{E}_{spat\_width} \in \mathbb{R}^{H\times(W-1)\times L}$. Both equations use HMSI, $\mathbf{Z}_{spat}$ as input. Moreover, the same operations are carried out for the reference images, HR-HSI, denoted as $\mathbf{Z}$.

$$\bar{\mathbf{E}}_{spat\_height}(i, j, k) = \mathbf{Z}(i+1, j, k) - \mathbf{Z}(i, j, k), \tag{3}$$

$$\bar{\mathbf{E}}_{spat\_width}(i, j+1, k) = \mathbf{Z}(i, j, k) - \mathbf{Z}(i, j, k), \tag{4}$$

After obtaining the results of the four equations, the HMSI edge map and the reference edge map are compared by mean squared error function (MSE)

$$\mathcal{L}_{spat\_height} = \frac{\sum_{k=1}^{L}\sum_{i=1}^{(H-1)}\sum_{j=1}^{W}(\mathbf{E}_{spat\_height}(i, j, k) - \bar{\mathbf{E}}_{spat\_height}(i, j, k))^2}{2WL(H-1)}, \tag{5}$$

$$\mathcal{L}_{spat\_width} = \frac{\sum_{k=1}^{L}\sum_{i=1}^{H}\sum_{j=1}^{(W-1)}(\mathbf{E}_{spat\_width}(i, j, k) - \bar{\mathbf{E}}_{spat\_height}(i, j, k))^2}{2HL(W-1)}, \tag{6}$$

$\mathcal{L}_{spat\_height}$ and $\mathcal{L}_{spat\_width}$ represent the loss function with respect to height and width respectively, which are combined into $\mathcal{L}_{spat}$ by the following equation

$$\mathcal{L}_{spat} = \mathcal{L}_{spat\_height} * 0.5 + \mathcal{L}_{spat\_width} * 0.5, \tag{7}$$

C. Spectral Module

This module takes as input the output of the previous module, $\mathbf{Z}_{spat}$ which it applies a convolutional layer SpecRN with the same characteristics as the one used in the spatial module, kernel set in 3×3 and 1 step, where is obtained the edge map of $\mathbf{Z}_{spat}$, $\mathbf{E}_{spec} \in \mathbb{R}^{H\times W\times(L-1)}$. In this part attention is paid to reconstructing the spectral information, for which a similar loss function is used as in SpatRN, $\mathcal{L}_{spec}$ in this occasion.

$$\mathbf{E}_{spec}(i, j, k) = \mathbf{Z}_{spec}(i, j, k+1) - \mathbf{Z}_{spec}(i, j, k), \tag{8}$$

$$\bar{\mathbf{E}}_{spec}(i, j, k) = \mathbf{Z}(i, j, k+1) - \mathbf{Z}(i, j, k), \tag{9}$$

$$\mathcal{L}_{spec} = \frac{\sum_{k=1}^{(L-1)}\sum_{i=1}^{H}\sum_{j=1}^{W}(\mathbf{E}_{spec}(i, j, k) - \bar{\mathbf{E}}_{spec}(i, j, k))^2}{2HW(L-1)}, \tag{10}$$

Where $\mathbf{E}_{spec} \in \mathbb{R}^{H\times W\times(L-1)}$ is the edge map from $\mathbf{Z}$. On this occasion, since we move across one dimension, the spectral bands (k), only one loss function is calculated.

At the end of the process, MSE function is used again between the reference HMSI and the HR-HSI edge maps.

Finally, $\mathbf{Z}_{spec}$ correspond with the estimated HR-HSI, which is optimized with the following loss function

$$\mathcal{L}_{opti} = \frac{\sum_{k=1}^{L}\sum_{i=1}^{H}\sum_{j=1}^{W}(\mathbf{Z}_{spec}(i,j,k) - \bar{\mathbf{Z}}(i,j,k))^2}{2HWL}, \tag{11}$$

where $\mathbf{Z}$ is the map edge of reference. The final loss function is represented as

$$\mathcal{L} = \mathcal{L}_{spat} + \mathcal{L}_{spec} + \mathcal{L}_{opti} \tag{12}$$

D. Total Variation (TV) loss function

TV-loss was proposed first time by [14], which describe the sum of the absolute differences between neighboring pixels, both horizontally and vertically. In this way TV-loss is able to attenuate the noise of the output image, since it measures the input noise and produces smoothness throughout the entire spatial dimension of the output image [12]. This algorithm studies the vertical and horizontal differences at the same time, unlike in the SSR-NET model, which uses two different functions to calculate the height and width differences independently, as it appears in the Equation 5 and Equation 6. For our experiments, TV-loss takes as input the output of the Spatial Module. TV-loss function is implemented as follows:

$$TV Loss = 2c \cdot \frac{1}{WH} \cdot \left(\frac{\sum_{ij}(x_{i,j+1} - x_{i,j})^2}{H} + \frac{\sum_{ij}(x_{i+1,j} - x_{i,j})^2}{W} \right) \tag{13}$$

where c is the TV-loss weight, x represent the pixel and H and W are the dimensions, of height and width respectively.

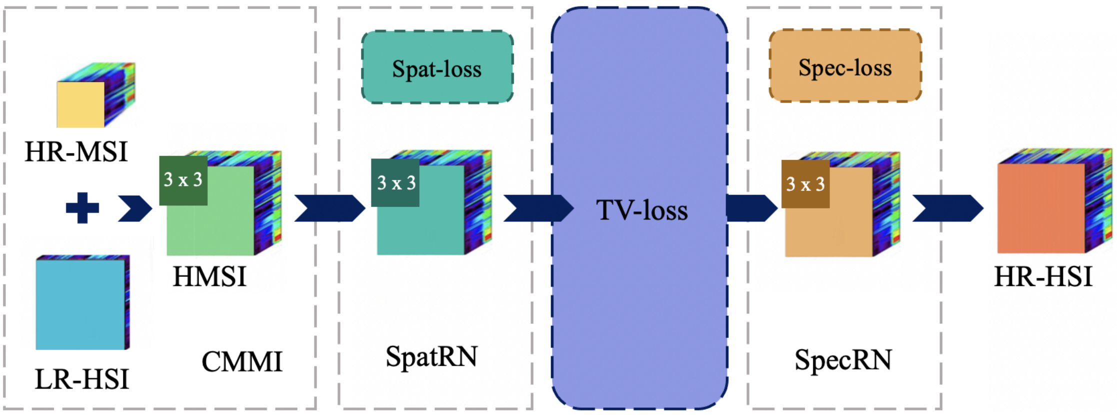

Figure 2: Framework used in the experiments to check the efficiency of TV-loss. TV-loss uses the output of the SpacRN module as input. The 3×3 squares indicate the convolutional layers that are applied to the input images. CMMI, SpatRN and SpecRN show the Cross-Mode Message Inserting, Spatial, and Spectral module respectively.

E. Smooth L1-loss function

As shown in the spatial module and spectral module of SSR-NET explained above, to compare the edge maps generated by the module with the reference edge maps, they use MSE. In the experiments in this paper, we replace MSE function with Smooth L1-loss, and see how it affects the network. Smooth L1-loss algorithm uses a criterion that when the absolute error of the elements is less than beta applies the square, otherwise apply L1 term [15]. This function is described as follows:

$$loss(x,y) = \frac{1}{n}\sum_{i} z_i \tag{14}$$

where z_i:

$$z_i = \begin{cases} 0.5(x_i - y_i)^2/beta, & if|x_i - y_i| < beta \\ |x_i - y_i| - 0.5 * beta, & otherwise \end{cases} \tag{15}$$

beta is a modifiable parameter that by default is equal to 1, which is the parameter to be used for the experiments in the experiments.

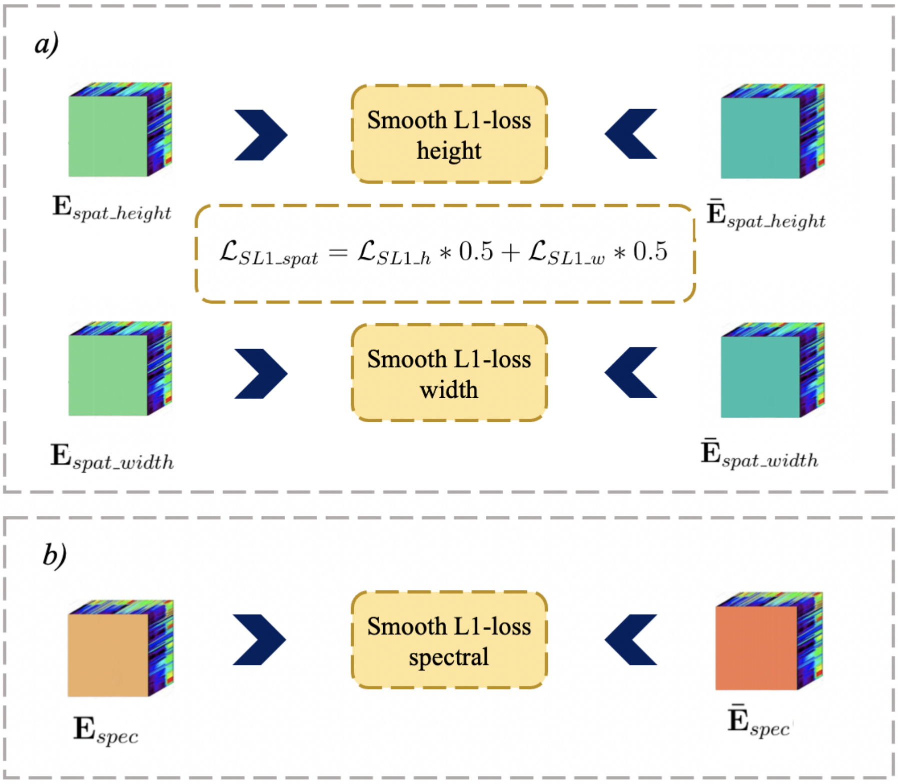

Figure 3: Smooth L1-loss implementation. a) Smooth L1-loss takes as input the features map estimated after the spatial module ($\mathbf{E}_{spat\_height}$ and $\mathbf{E}_{spat\_width}$), and the reference map of HR-HSI ($\bar{\mathbf{E}}_{spat\_height}$ and $\bar{\mathbf{E}}_{spat}$) to calculate the loss. To obtain the final loss, the results are used after applying the Smooth L1-loss function along the height and width. b) In the spectral module, the features maps of the estimated images ($\mathbf{E}_{spec}$) are utilized with respect to the reference ($\bar{\mathbf{E}}_{spec}$) ones along the spectral dimension.

3 Experiments

In this section we begin by exposing the five different databases used to carry out the experiments, in addition to the metrics applied to support the results. The section continues with the experimental settings, and finally concludes with the results obtained from the implementations respect to the original SSR-NET results.

3.1 Data Sets

The databases that are going to be exposed below are available as an open resource [16], in addition the images have their corresponding ground truth.

1) Botswana: this database was captured in 2001-2004 by the Hyperion sensor of the NASA EO-1 satellite over the Okavango Delta. The data consist of 147 x 256 pixels with a spatial resolution of 30 m. Spectral bands cover wavelengths from 400 to 2500 nm. Removing the uncalibrated and noisy bands of water absorption features 145 bands left.

2) Urban: urban dataset was collected in 1995 over Copperas Cove, TX, USA. There are 307 x 307 pixels with 2 m as spatial resolution. Furthermore, is composed of 210 bands in total in a range from 400 to 2500 nm with an interval of 10 nm. Urban is made up of captures of buildings, architectural structures or urban landscapes. When removed bands of dense water vapor and atmospheric, are obtained 162 bands.

3) Pavia Center: the Pavia Center database was collected with the same optical sensor used to capture the images from Pavia University, thus having the same spatial resolution, 1.3 m. However, in this case each band has 1096 x 1096 pixels.

4) Pavia University: database obtained in 2003 by ROSIS-3 optical airborne sensor over the area of the University of Pavia, Italy. This dataset has 610 x 610 pixels and 1.3 m resolution. The bands are 115 with a spectral range of 430 to 860 nm within an interval of 10 nm.

5) Kennedy Space Center (KSC): This database is collected with the AVIRIS optical sensor from NASA, on March 23, 1996, over the Kennedy Space Center (KSC) in Florida. The images have 224 bands with a range of 400 to 2500 nm in 10 nm steps and a spatial resolution of 18 m. After removing the water absorption bands, 176 bands remain.

3.2 Evaluation Metrics

To analyze the improvement of our implementations, we have used four different metrics applied to each data set. These metrics evaluate both the spatial and spectral quality of the estimated images after the network, compared to their respective ground truth.

1) Peak Signal-to-Noise Ratio (PSNR): the peak SNR (PSNR) is a very popular quality metric, used to evaluate the spatial quality of the bands in the reconstructed HR-HSI. This metric measures the similarities between the fused image and the reference image.

$$PSNR = 10 \log_{10} \left(\frac{\max(\mathbf{R}_k)^2}{\frac{1}{HW} \parallel \mathbf{R}_k - \mathbf{Z}_k \parallel_2^2} \right), \tag{16}$$

where R and Z are the *kth* band of the reference and estimated fused image, respectively. The final result is the average of all the bands, where a higher result indicates a better spatial quality of the fused image.

2) Erreur Relative Globale Adimensionelle de Synthèse (ERGAS): the ERGAS measures the quality of the fused image in a global statistical way, where the ideal value would be 0.

$$ERGAS = \frac{100}{r}\sqrt{\frac{1}{L}\sum_{k=1}^{L}\frac{\| \mathbf{R}_k - \mathbf{Z}_k \|_2^2}{\mu^2(\mathbf{R}_k)}}, \tag{17}$$

where r is the ratio of the spatial resolution from HR-MSI to LR-HSI. R_k and Z_k denotes the *kth* bands of the reference and fused image, accordingly. Moreover, $\mu(R_k)$ represents the mean value of the *kth* band in the reference image.

3) Spectral Angle Mapper (SAM): this metric evaluates the spectral information preservation at each pixel, important to estimate the spectral distortion.

$$SAM = \arccos\left(\frac{\langle \mathbf{R}(i,j), \mathbf{Z}(i,j)\rangle}{\| \mathbf{R}(i,j) \|_2 \| \mathbf{Z}(i,j) \|_2}\right), \tag{18}$$

where $\mathbf{R}(i,j)$ and $\mathbf{Z}(i,j)$ represent the spectral vectors at the spatial pixel position (i,j) in the reference and estimated fused image respectively. In addition, $\langle \mathbf{R}(i,j), \mathbf{Z}(i,j)\rangle$ is the two vector inner product. The final result is achieved by averaging the SAM metric for all pixels. A better spatial quality is obtained for SAM values close to 0.

4) Root-Mean-Squared Error (RMSE): the RMSE measures the difference between the estimated and reference image, to compare the prediction errors.

$$RMSE = \sqrt{\frac{\sum_{k=1}^{L}\sum_{i=1}^{H}\sum_{j=1}^{W}(\mathbf{R}_k(i,j) - \mathbf{Z}_k(i,j))^2}{HWL}}, \tag{19}$$

The best results are obtained with smaller values for the RMSE.

3.3 Experimental settings

For the experimental results we have used the same ones that appear in [9]. In the databases, the central region of dimensions 128 x 128 was cropped, which is used for the test, while another region of the same size is randomly cropped to use it in the training.

To evaluate how the implementations of the proposed algorithms affect, experiments have been carried out on three different networks, one with the implementation of TV-loss, another with Smooth L1-loss, and the last network with both loss functions combined. For the training stage Adam works as an optimizer and all models are trained with 10000 iterations.

We have done the experiments on Python 2.7.17. The computer used is Alienware Aurora r8, Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz, with RAM of 32GB and the GPU GeForce GTX 2080, 11GB.

3.4 Comparison with Data Sets

A) Botswana Data Set: for this database, the implementation of TV-loss together with Smooth L1-loss gets excellent results for the ERGAS metric, achieving a great improvement compared to the original SSR-Net. These results represent an improvement in global information with our implementation, since ERGAS evaluates this aspect. TV-loss plus Smooth L1-loss manage to solve the problem that spatial edge loss and spectral edge loss concentrate on local features [9] which obtain poor results in the global evaluation. These good results in the experiments are achieved with very close values for PSNR in SSR-Net, and getting better results for the SAM metric as well. In Figure 4 the good results of ERGAS are shown without damaging the PSNR, during the training stage.

Botswana				
Method	PSNR	RMSE	ERGAS	SAM
TV-loss	35.6858243	0.5984603	9.8286364	2.9243915
Smooth L1-loss	35.3552605	0.6216752	9.4089469	2.8217714
TV + Smooth L1	35.9895400	0.5778959	7.8465301	2.6916665
SSR-Net	36.0953531	0.5708985	9.0349448	2.7552987

Table 1: Quantitative comparison of TV-loss, Smooth L1-loss and both added implementations, with respect to the original SSR-Net [9]. The best mark appears in RED, and the second best mark in GREEN.

B) Urban Data Set: in the images from the Urban database, the TV-loss implementation achieves the best marks for all metrics, followed by the Smooth L1-loss implementation which achieves the second best marks, except for the SAM metric, which does not obtain good spectral information preservation, being better for SSR-Net. However, for this database, the implementation of TV-loss plus Smooth L1-loss scores below SSR-Net.

Urban				
Method	PSNR	RMSE	ERGAS	SAM
TV-loss	36.5380089	2.9326284	1.4765417	2.7778814
Smooth L1-loss	36.3140211	3.0092373	1.5204431	2.8409947
TV + Smooth L1	33.2532634	4.2804980	2.0310636	2.8240672
SSR-Net	35.5324254	3.2925792	1.6043167	2.7424033

Table 2: Quantitative comparison of TV-loss, Smooth L1-loss and both added implementations, with respect to the original SSR-Net [9]. The best mark appears in RED, and the second best mark in GREEN.

C) Pavia Center Set and Pavia University Data Set: since these databases are similar, we obtain similar data for the different implementations, with TV-loss being the implementation that achieves the best results. For Pavia Center, TV-loss it obtains the best marks for all metrics, while in Pavia University it gets the best value in SAM, and the best second values for the rest of the metrics. On the other hand, Smooth L1-loss and TV-loss plus Smooth L1-loss implementations generally fail to improve SSR-Net results in these set of images.

Pavia Center				
Method	PSNR	RMSE	ERGAS	SAM
TV-loss	37.5938442	3.3639379	3.8563237	3.8283927
Smooth L1-loss	36.5397383	3.7979850	4.3929020	3.9856211
TV + Smooth L1	36.4896502	3.8199499	4.3908384	4.0894233
SSR-Net	37.3523484	3.4587786	3.9763441	3.8706276

Table 3: Quantitative comparison of TV-loss, Smooth L1-loss and both added implementations, with respect to the original SSR-Net [9]. The best mark appears in RED, and the second best mark in GREEN.

Pavia University				
Method	PSNR	RMSE	ERGAS	SAM
TV-loss	41.4144806	2.0895948	1.4395748	2.0651369
Smooth L1-loss	41.1734903	2.1483824	1.4492738	2.0935716
TV + Smooth L1	41.0765007	2.1725063	1.5012223	2.1848737
SSR-Net	42.0396468	1.9444813	1.3698532	2.2337068

Table 4: Quantitative comparison of TV-loss, Smooth L1-loss and both added implementations, with respect to the original SSR-Net [9]. The best mark appears in RED, and the second best mark in GREEN.

D) Kennedy Space Center Data Set: for the latest database, Kennedy Space Center, it is shown once again how the better marks are taken by implementations, highlighting the results generated by TV-loss plus Smooth L1-loss for these image collection. TV-loss plus Smooth L1-loss gets top marks for all metrics, as in Botswana data set. Both, KSC and Botswana, are databases with high pixel spatial resolution, so from the results it can be deduced that for high resolution images, the implementation of TV-loss plus Smooth L1-loss is the most suitable option for the image fusion. Moreover, the implementation of Smooth L1-loss has good results as well, getting the second best marks for the ERGAS and SAM metrics, although for the PSNR the second best result is obtained by SSR-Net, as expected, since Mean Square Error manages to improve the PSNR, and in SSR-Net it is the method used to calculate the loss.

Kennedy Space Center				
Method	PSNR	RMSE	ERGAS	SAM
TV-loss	24.0585083	15.981398	530.905434	51.0625010
Smooth L1-loss	23.9601858	16.163332	203.169304	23.3826027
TV + Smooth L1	24.2364655	15.657301	203.019369	23.3345729
SSR-Net	23.8550292	16.360205	548.426875	52.4396072

Table 5: Quantitative comparison of TV-loss, Smooth L1-loss and both added implementations, with respect to the original SSR-Net [9]. The best mark appears in RED, and the second best mark in GREEN.

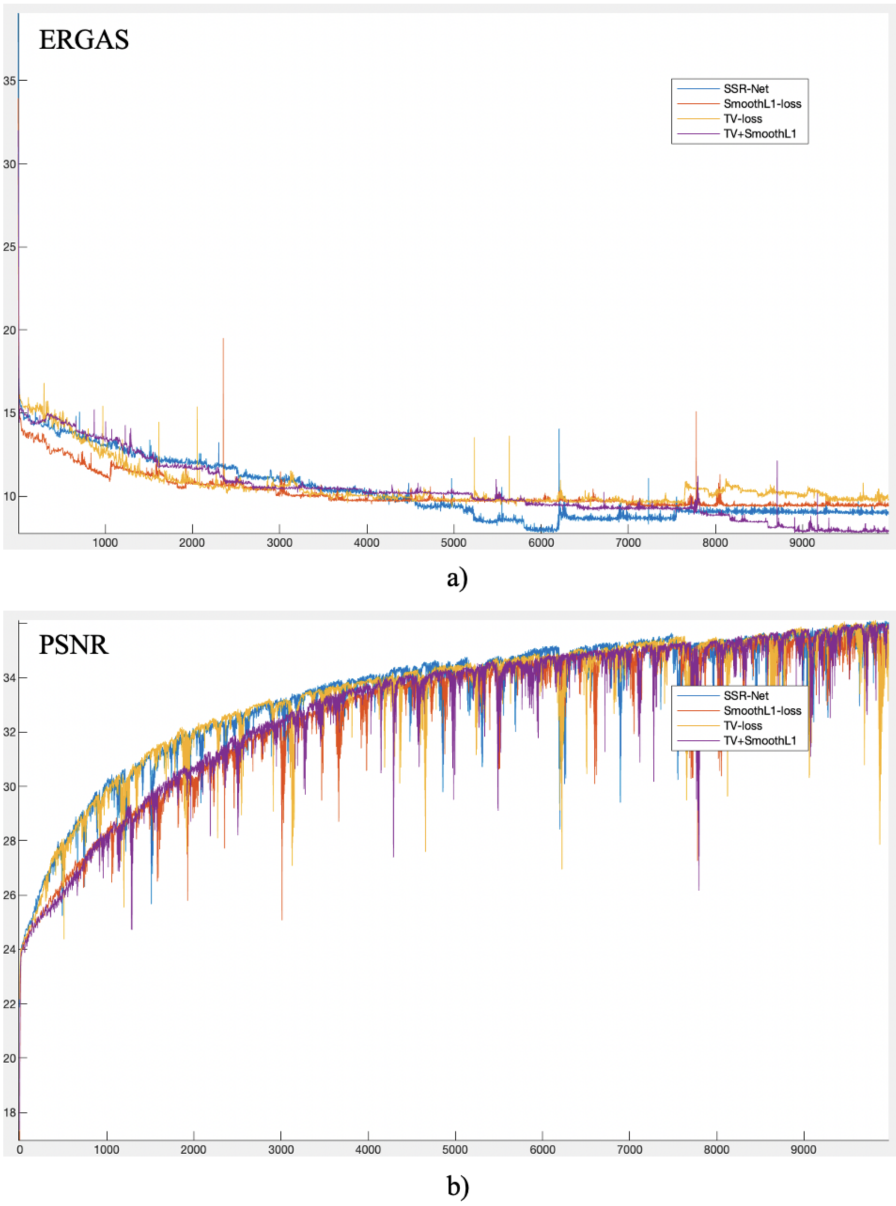

Figure 4: Metrics (a) ERGAS and (b) PSNR during the training stage for each of the implementations and for the original SSR-Net [9]. The x-axis indicates the number of iterations.

4 Conclusion

In this paper, the implementations of TV-loss, Smooth L1-loss and both loss functions combined, are proposed to achieve improved results in multispectral and hyperspectral image fusion. After the experiments in the five databases selected, we observe the implementations represent an improvement in the results with respect to the original SSR-Net chosen as reference. The implementation of TV-loss plus Smooth L1-function manages to improve the fusion for the Botswana and Kennedy Space Center databases, indicating that TV-loss plus Smooth L1-loss improves extraction of complex spatial features, since the two databases have a high spatial resolution per pixel, 30 m and 18 m respectively. We especially get excellent results for the metric that assesses fusion quality globally (ERGAS). This implementation manages to provide a global restoration in the final HR-HSI without excessively damaging the PSNR, solving the problem that spatial edge loss and spectral edge loss only focus on the restoration of local features. On the other hand, for databases with less resolution, such as Pavia Center, Pavia University or Urban, we obtain the TV-loss is the best of the implementations, getting the best results for all the metrics in both Urban and Pavia Center data sets. The reason why implementation of the two loss functions combined worsens the results in these databases, it may be because both algorithms together overly softens the spatial features of the images with less spatial resolution, since the two algorithms focus on denoising.

Summing up, the implementations improve the results, being an interesting proposal to continue investigating in the different combinations between the TV-loss and Smooth L1-loss functions to improve the hyperspectral image restoration. As future works, it can be tested with TV-loss along the spectral dimension, since in our article TV-loss focuses on spatial features. After the good results obtained from the TV-loss implementation, another important future work would be the incorporation of a graph total variation (GTV) that supports our TV-loss function.

5 Acknowledgements

Above all, thanks to my supervisors Hao Wang and Guoxia Xu, for having guided and solved the problems that have arisen throughout this project, as well as providing the necessary knowledge to be able to successfully complete the results. And also thanks to the NTNU Computer Science department for supplying the necessary GPU and computer to carry out the experiments, especially to Binu Melit Devassy the person who has managed the access to the machine.

Bibliography

[1] R. Dian, S. Li, A. Guo and L. Fang, 'Deep hyperspectral image sharpening', *IEEE transactions on neural networks and learning systems*, no. 99, pp. 1–11, 2018.

[2] R. Dian, S. Li, B. Sun and A. Guo, 'Recent advances and new guidelines on hyperspectral and multispectral image fusion', *Information Fusion*, 2020.

[3] Z. Shao and J. Cai, 'Remote sensing image fusion with deep convolutional neural network', *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 11, no. 5, pp. 1656–1669, 2018.

[4] J. Yang, Y.-Q. Zhao and J. C.-W. Chan, 'Hyperspectral and multispectral image fusion via deep two-branches convolutional neural network', *Remote Sensing*, vol. 10, no. 5, p. 800, 2018.

[5] X.-H. Han, Y. Zheng and Y.-W. Chen, 'Multi-level and multi-scale spatial and spectral fusion cnn for hyperspectral image super-resolution', in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[6] G. Masi, D. Cozzolino, L. Verdoliva and G. Scarpa, 'Pansharpening by convolutional neural networks', *Remote Sensing*, vol. 8, no. 7, p. 594, 2016.

[7] F. Palsson, J. R. Sveinsson and M. O. Ulfarsson, 'Multispectral and hyperspectral image fusion using a 3-d-convolutional neural network', *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 639–643, 2017.

[8] R. Dian, S. Li, A. Guo and L. Fang, 'Deep hyperspectral image sharpening', *IEEE transactions on neural networks and learning systems*, no. 99, pp. 1–11, 2018.

[9] X. Zhang, W. Huang, Q. Wang and X. Li, 'Ssr-net: Spatial-spectral reconstruction network for hyperspectral and multispectral image fusion', *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[10] W. He, H. Zhang, L. Zhang and H. Shen, 'Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration', *IEEE transactions on geoscience and remote sensing*, vol. 54, no. 1, pp. 178–188, 2015.

[11] H. K. Aggarwal and A. Majumdar, 'Hyperspectral image denoising using spatio-spectral total variation', *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 3, pp. 442–446, 2016.

[12] A. Chadha, J. Britto and M. M. Roja, 'Iseebetter: Spatio-temporal video super-resolution using recurrent generative back-projection networks', *Computational Visual Media*, vol. 6, no. 3, pp. 307–317, 2020.

[13] W. Chlewicki, F. Hermansen and S. Hansen, 'Noise reduction and convergence of bayesian algorithms with blobs based on the huber function and median root prior', *Physics in Medicine & Biology*, vol. 49, no. 20, p. 4717, 2004.

[14] H. A. Aly and E. Dubois, 'Image up-sampling using total-variation regularization with a new observation model', *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1647–1659, 2005.

[15] SmoothL1Loss. (). 'Smoothl1loss - pytorch 1.7.0 documentation', [Online]. Available: https://pytorch.org/docs/stable/generated/torch.nn.SmoothL1Loss.html#torch.nn.SmoothL1Loss (visited on 07/12/2020).

[16] H. R. S. Scenes. (). 'Hyperspectral remote sensing scenes - grupo de inteligencia computacional (gic)', [Online]. Available: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes (visited on 07/12/2020).