Southwest Jiaotong University

HYPERSPECTRAL IMAGE CLASSIFICATION BASED ON TENSOR-TRAIN CONVOLUTIONAL LONG SHORT-TERM MEMORY

Wen-Shuai Hu, Heng-Chao Li, Tian-Yu Ma, Qian Du,
Antonio Plaza, Willian J. EMERY

Email：lihengchao_78@163.com

Contents

1 Introduction

1.1 Background

✓ **Hyperspectral remote sensing Image**

a) has been widely applied in various fields.

b) is naturally a cube.

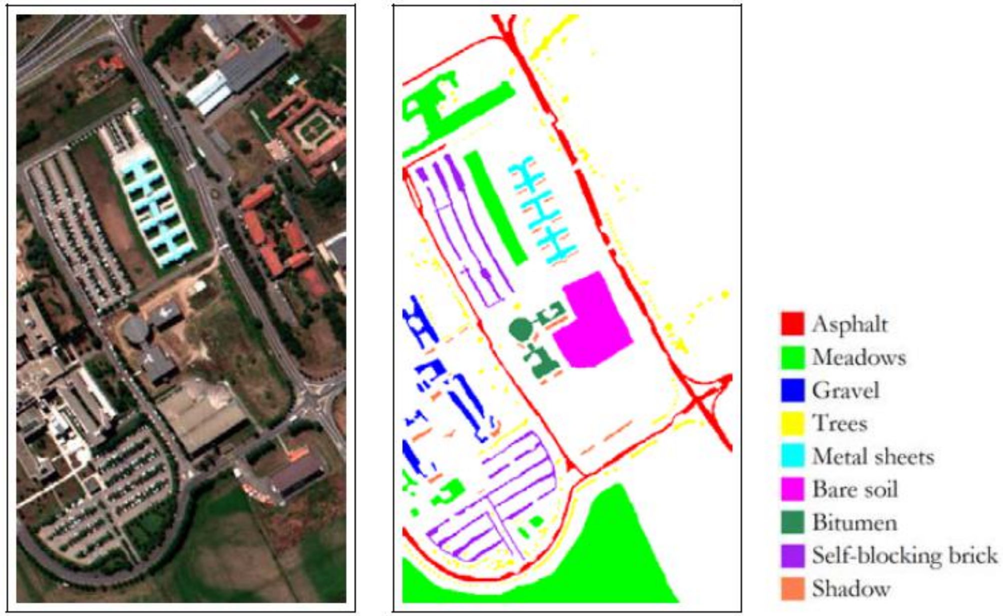
c) high-dimensional leads to information redundancy.

1 Introduction

1.1 Background

✓ **Hyperspectral Classification**

- Given a set of observations (pixel vectors), the goal of classification is to assign a unique label to each pixel vector so that it is uniquely represented by a given class.

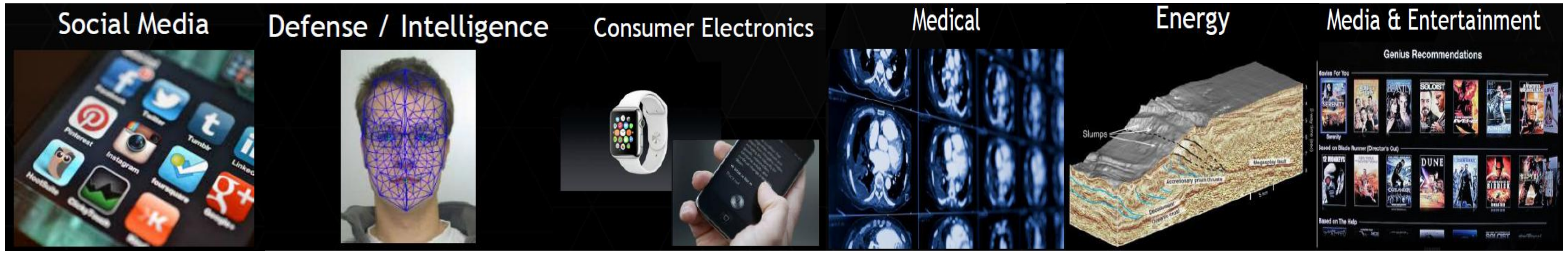


1 Introduction

1.2 Deep learning fundamentals

✓ Convolutional Long Short-Term Memory (ConvLSTM)

- Deep learning is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms.

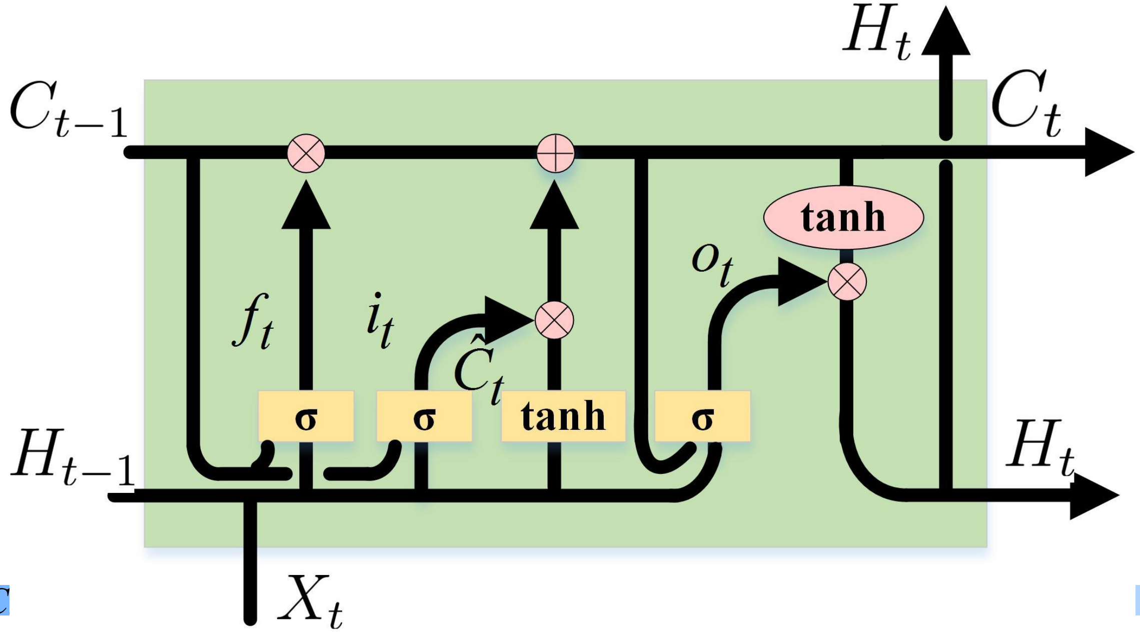


1 Introduction

1.2 Deep learning fundamentals

✓ Convolutional Long Short-Term Memory (ConvLSTM)

- ConvLSTM (actually 2-D version of LSTM, ConvLSTM2D) is realized by extending 1-D data processing method in LSTM to the 2-D convolution operation, which enables it to capture the spatial information and model the long-range dependencies from input data, as shown in Fig. 1.



✓Fig. 1 The inner structure of ConvLSTM cell (actually ConvLSTM2D cell)

1 Introduction

1.2 Deep learning fundamentals

✓ **Convolutional Long Short-Term Memory (ConvLSTM)**

- There are two kinds of convolutional weights, such as $W_{x.}$ with a size of k × k × C × S and $W_{h.}$ with a size of k × k × S × S (ignoring Hadamard product). Therefore, the total number of the parameters in each ConvLSTM2D cell is

$$N = 4k^2 S(S+C)$$

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i)$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f)$$

$$\tilde{C}_t = \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t$$

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \circ C_t + b_o)$$

$$H_t = o_t \circ \tanh(C_t)$$

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

✓ Tensor-Train decomposition (TTD)

- Tensor-train decomposition (TTD) is an effective tensor factorization algorithm.
- Given a d-dimensional tensor $\mathcal{A} \in R^{l_1 \times l_2 \ldots \times l_d}$, let $l_d = m_d \cdot n_d$, $\mathcal{A}$ is rewrited as $\mathcal{A} \in R^{(m_1 \cdot n_1) \times (m_2 \cdot n_2) \ldots \times (m_d \cdot n_d)}$. By using TTD, the factorization
- of $\mathcal{A}$ can be expressed as

$$\mathcal{A}((i_1, j_1), (i_2, j_2), \ldots, (i_d, j_d)) = \mathcal{G}'_1[i_1, j_1]\mathcal{G}'_2[i_2, j_2] \ldots \mathcal{G}'_d[i_d, j_d],$$

- where $\mathcal{G}'_k[i_k, j_k] \in R^{r_{k-1} \times r_k}$, and the collection of $\left\{\mathcal{G}'_k\right\}_{k=1}^{d}$ is called $TT - cores$ The set of the elements $\{r_k\}_{k=0}^{d}$ is $TT - ranks$, and the values of r_0 and r_d are 1.

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

✓ **Tensor-Train convolutional layer (TTC layer)**

- For basic convolutional layer in CNN, the input $\mathcal{X} \in R^{W \times H \times C}$ is transformed into the output $\mathcal{Y} \in R^{W \times H \times S}$ by a convolution kernel $\mathcal{K} \in R^{l \times l \times C \times S}$
- By decomposing it along the channel dimension with TTD, a TT-convolutional layer (as shown Fig.2) is built, whose outputs can be described as

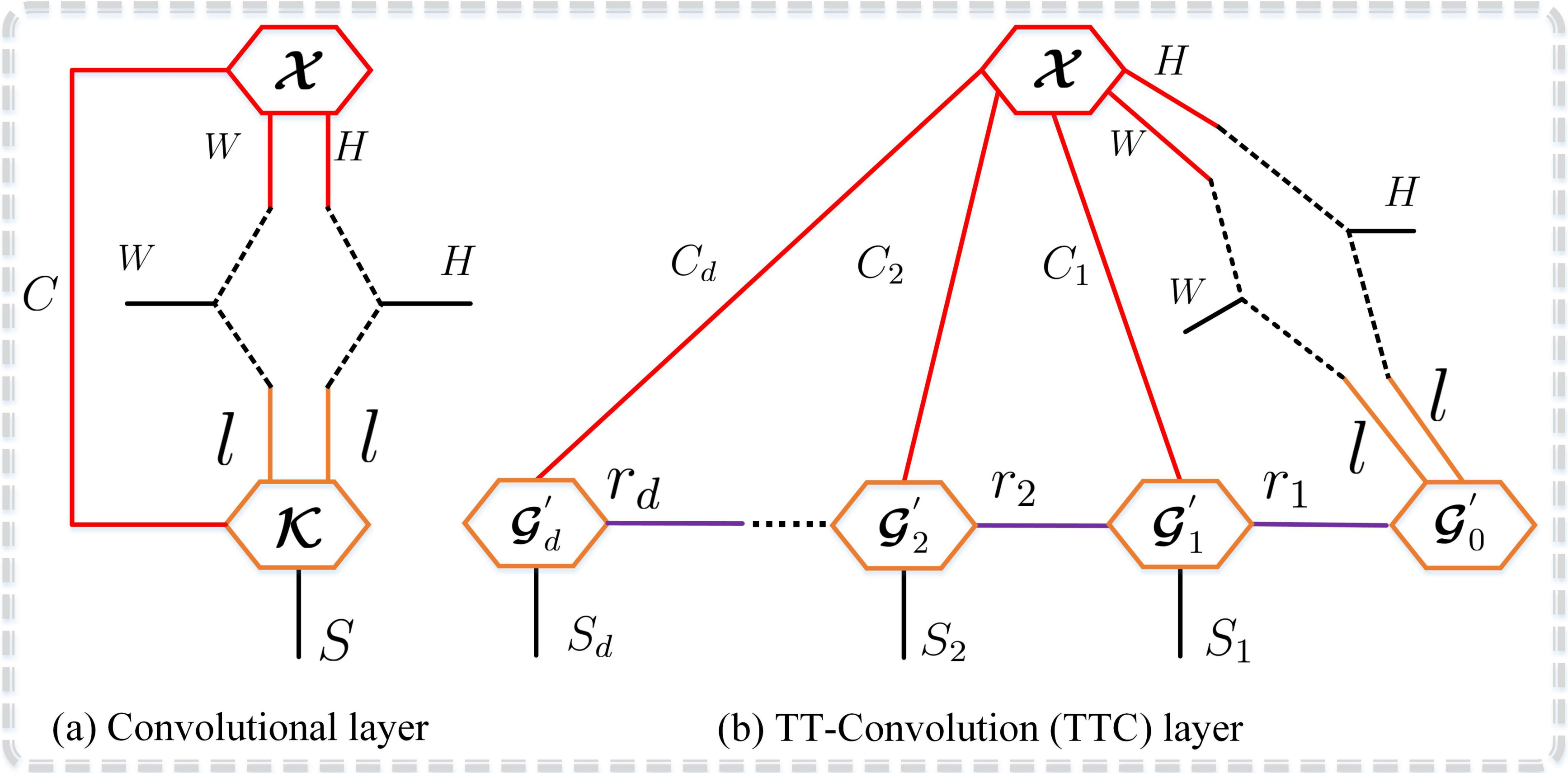
$$\mathcal{Y}(x, y, s_1, s_2, \ldots, s_d) = \sum_{i=1}^{l} \sum_{j=1}^{l} \sum_{c_1, c_2, \ldots, c_d} \mathcal{X}(i + x - 1, j + y - 1, c_1, c_2, \ldots, c_d) \cdot \mathcal{G}_0'[i, j] \mathcal{G}_1'[c_1, s_1] \mathcal{G}_2'[c_2, s_2] \ldots \mathcal{G}_d'[c_d, s_d],$$

- where $C = \prod_{k=1}^{d} C_k$, $S = \prod_{k=1}^{d} S_k$, $\mathcal{G}_0'[i, j]$ and $\mathcal{G}_k'[c_k, s_k]$ are the $TTC - cores$ $(k = 1, 2, \ldots, d)$.

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

✓ **Tensor-Train convolutional layer (TTC layer)**



✓Fig. 2 Illustration of the TTC layer

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

✓ **Tensor-Train Convolutional Long Short-Term Memory (TT-ConvLSTM2D)**

- Inspired by the above, a lightweight ConvLSTM2D cell is built by using TTD, namely TT-ConvLSTM2D cell, in which the decompositions of $W_{x\cdot}$ and $W_{h\cdot}$ corresponding to input and output gate can be written as

$$\boldsymbol{\mathcal{W}}_{x\cdot} = \boldsymbol{\mathcal{G}}_0^{'}[i,j]_{x\cdot}\boldsymbol{\mathcal{G}}_1^{'}[c_1,s_1]_{x\cdot}\boldsymbol{\mathcal{G}}_2^{'}[c_2,s_2]_{x\cdot}\ldots\boldsymbol{\mathcal{G}}_d^{'}[c_d,s_d]_{x\cdot}$$
$$\boldsymbol{\mathcal{W}}_{h\cdot} = \boldsymbol{\mathcal{G}}_0^{'}[i,j]_{h\cdot}\boldsymbol{\mathcal{G}}_1^{'}[s_1,s_1]_{h\cdot}\boldsymbol{\mathcal{G}}_2^{'}[s_2,s_2]_{h\cdot}\ldots\boldsymbol{\mathcal{G}}_d^{'}[s_d,s_d]_{h\cdot}$$

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

✓ **Tensor-Train Convolutional Long Short-Term Memory (TT-ConvLSTM2D)**

- The total number of the parameters in each TT-ConvLSTM2D cell is

$$N_2 = 8k^2 r_1 + 4\sum_{j=1}^{d} s_j r_{j+1} r_j (c_j + s_j).$$

- The compression rate of each TT-ConvLSTM2D cell is described as

$$\frac{N_1}{N_2} = \frac{k^2 S(S+C)}{2k^2 r_1 + \sum_{j=1}^{d} s_j r_{j+1} r_j (c_j + s_j)}.$$

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

- *Step 1: Data preparation*

The principal component analysis (PCA) is applied to perform a preprocessing, and the first K principal components are selected as a input image so as to reduce spectral redundancy.

Combined with spatial information of each pixel with a size of $s \times s$, a 3-D data represented as $X^H \in R^{s \times s \times K}$ is built from original HSI data as the input image.

2 TT-ConLSTM2D for HSI Classification

2.1 Propose framework

- *Step 2: Construct Spectral sequence(cont.)*

To adapt to the requirement of the developed TT-ConvLSTM2D layer, X^H is further decomposed into τ 2-D components, then converted into a sequence

$$X^H \Rightarrow \{X_1^H, \ldots, X_t^H, \ldots, X_\tau^H\}, t \in \{1, 2, \ldots, \tau\}$$

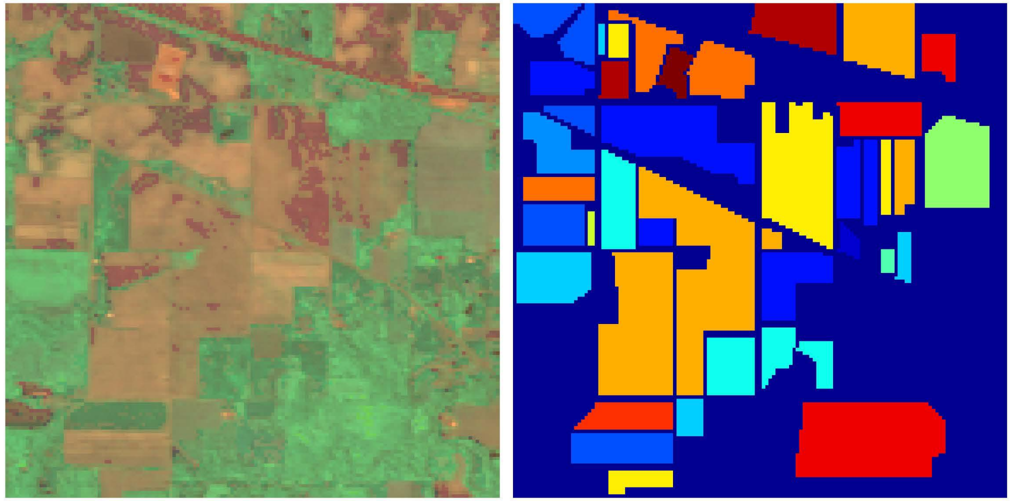
where τ is the dimension $time\_step$ in each TT-ConvLSTM2D layer. τ equals K in this paper. $X_t^H \in R^{s \times s}$ is the input of each TT-ConvLSTM2D cell.

3 Experiments

3.1 Experiment Data and Setup(*cont.*)

The experiments are conducted on Indian Pines data, which was acquired by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over the Indian Pines test site in northwest Indiana in 1992. It contains 16 different classes and a total of 10249 ground-truth samples (as shown in Fig.3).

Specifically, 10% of the whole samples are randomly chosen for training, and the rest for testing.



✓Fig. 3 The False-color map and ground truth map of Indian Pines

3 Experiments

3.1 Experiment Data and Setup(*cont.*)

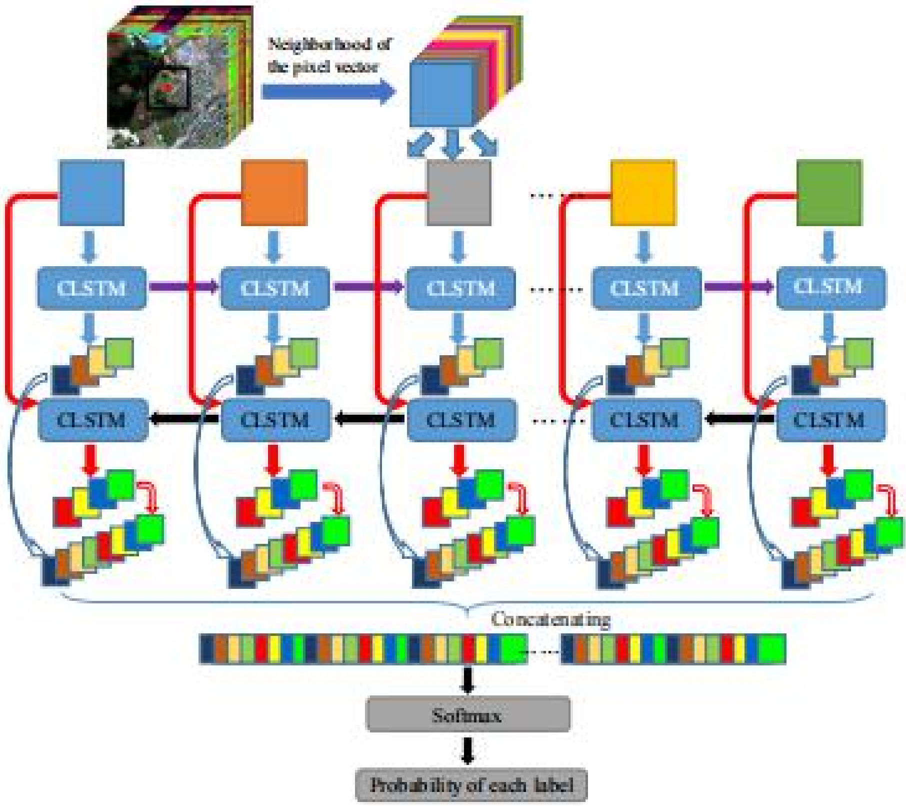
Two state-of-the-art ConvLSTM2D-based models are selected as the comparative methods, such as Bi-CLSTM[1], SSCL2DNN[2], to verify the superiority of our developed TT-ConvLSTM2D cell.

To eliminate the bias introduced by randomly choosing training samples, each experiment is repeated 10 times, and the mean values of each evaluation criterion are presented in the paper.

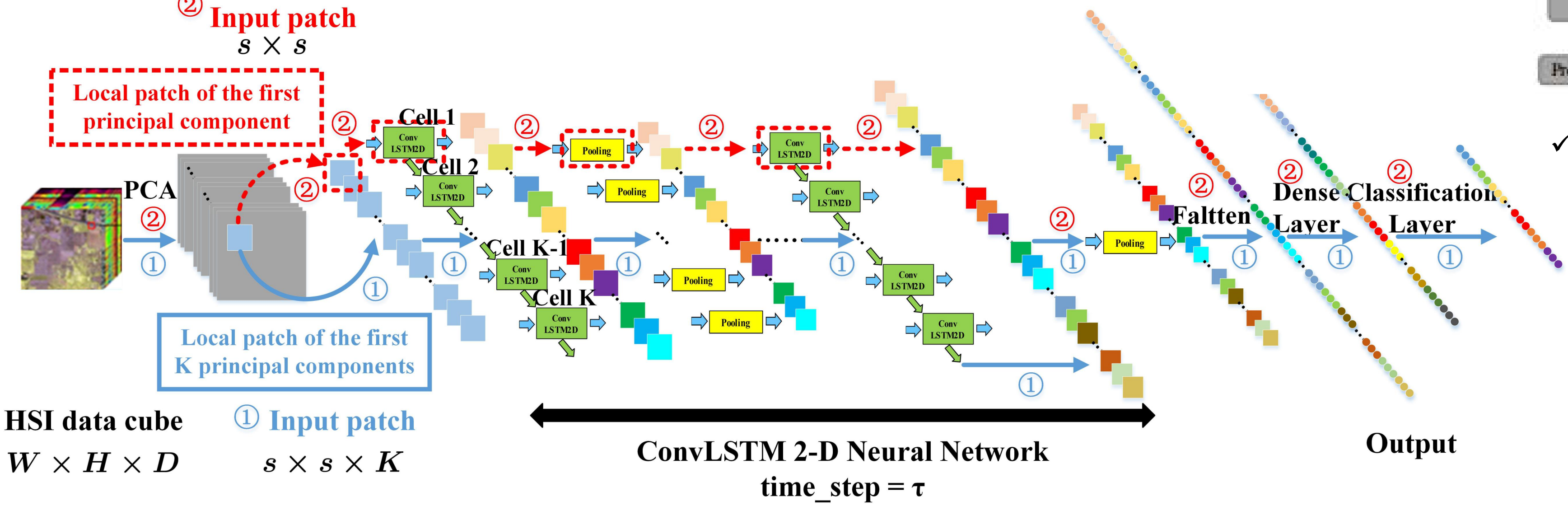
[1] Q. Liu, F. Zhou, R. Hang, and X. Yuan, ``Bidirectional-convolutional LSTM based spectral-spatial feature learning for hyperspectral image classification," Remote Sens., vol. 9, no. 12, pp. 1330, Dec. 2017.
[2] W. Hu, H Li, L. Pan, W. Li, R. Tao, and Q. Du, ``Spatial-spectral feature extraction via deep ConvLSTM neural networks for hyperspectral image classification," IEEE Trans. Geosci. Remote Sens., vol. 58, no. 6, pp.4237-4250, Jun. 2020.

3 Experiments

3.1 Experiment Data and Setup(*cont.*)



✓Fig. 4 Bi-CLSTM



✓Fig. 5 SSCL2DNN

3 Experiments

3.2 Experiment Results (cont.)

Table I gives the number of parameters and the compression rate in all ConvLSTM2D or TT-ConvLSTM2D layers.

✓Table I. NUMBER OF PARAMETERS AND COMPRESSION RATE

Model	Backbone	Number	Rate	OA
Bi-CLSTM	two parallel 3×3×32 layers two maxpool layers	760320	1	95.62%
Bi-TTCLSTM		**687360**	**1.106 ×**	**95.08%**
SSCL2DNN	a 4× 4 ×32 layer a 3 × 3 ×64 layer two maxpool layers	1443840	1	98.03%
SSTTCL2DNN		**253760**	**5.690 ×**	**97.33%**

3 Experiments

3.2 Experiment Results (cont.)

Table II lists the classification accuracy per class, as well as overall accuracy (OA), average accuracy (AA), and Kappa Coefficient (κ) .

Table I. Classification Results of Different Methods for The Indian Pines Data Set Using 10% Training Samples with $TTC - rank = 8$

Class	Bi-CLSTM	Bi-TTCLSTM	SSCL2DNN	SSTTCL2DNN
1	91.06	94.31	**100.00**	98.37
2	94.29	93.90	**98.11**	97.02
3	93.13	91.03	96.56	**96.74**
4	88.89	90.61	**96.56**	95.62
5	94.25	93.95	96.09	**96.32**
6	99.49	98.78	98.02	**98.83**
7	93.33	66.67	84.00	**92.00**
8	99.46	**99.69**	**99.69**	98.84
9	38.89	40.74	55.56	**81.48**
10	95.73	96.19	**97.07**	95.62
11	96.60	96.50	**99.34**	98.43
12	89.37	85.83	96.75	**96.82**
13	95.65	96.56	**98.19**	91.67
14	**99.80**	98.95	99.65	98.18
15	97.12	95.77	**98.85**	97.98
16	89.29	85.32	85.32	**92.46**
OA	95.62	95.08	**98.03**	97.33
AA	90.90	89.05	93.37	**95.40**
κ	94.99	94.37	**97.40**	96.96

4 Conclusion

- In this paper, we developed a lightweight TT-ConvLSTM2D cell by using tensor-train decomposition and introduced it for hyperspectral image classification.

- Experimental results demonstrate that the proposed TT-ConvLSTM2D cell can effectively reduce the number of the parameters of the whole models within a small range of accuracy degradation for hyperspectral image classification.

Thanks for your attention !