Progressive Band Processing of Fast Iterative Pixel Purity Index for Finding Endmembers

Chein-I Chang, *Life Fellow, IEEE*, Yao Li, and Yulei Wang

***Abstract*— This letter develops a progressive band processing (PBP) of fast iterative pixel purity index (FIPPI) according to a band sequential acquisition format in such a way that FIPPI can be processed band by band, while band acquisition is ongoing. As a result, PBP-FIPPI can generate progressive profiles of interband changes among PPI counts which allow users to observe significant bands that capture PPI counts. The idea to implement PBP-FIPPI is to use an inner loop specified by skewers and an outer loop specified by bands to process FIPPI. Interestingly, these two loops can also be interchanged with an inner loop specified by bands and an outer loop iterated by growing skewers. The resulting FIPPI is called progressive skewer processing of FIPPI. It turns out that both versions provide different insights into the design of FIPPI.**

***Index Terms*— Fast iterative pixel purity index (FIPPI), PBP of FIPPI (PBP-FIPPI), pixel purity index (PPI), progressive band processing (PBP), progressive skewer processing of FIPPI (PSP-FIPPI).**

I. Introduction

PIXEL purity index (PPI) [1] is one of the most widely used methods for finding endmembers [2]–[6]. However, there are also several major issues when PPI is implemented. First of all, it is sensitive to input parameters, namely, n_{skewer} (number of skewers) and t (cutoff threshold value). Second, the use of so-called skewer vectors (or skewers) which are randomly generated produces inconsistent final results. Third, it requires human intervention to manually select a final set of endmembers by visual inspection. To address the aforementioned issues a fast iterative algorithm, referred to as fast iterative pixel purity index (FIPPI) was developed by Chang and Plaza [3]. It has several significant advantages over PPI. It takes advantage of a recently developed concept, virtual dimensionality (VD) developed in [7] and [8] to estimate the initial number of skewers as endmembers for FIPPI. As a result, there is no need of using the two parameters, k and t for PPI. Furthermore, FIPPI makes use of automatic target generation process (ATGP) in [9] to generate an appropriate set of initial endmembers to initialize FIPPI that can speed up the algorithm considerably. Since the initial endmembers are specified by ATGP-generated targets, there is no issue of randomness resulting from skewers.

Manuscript received April 10, 2017; accepted May 25, 2017. Date of publication July 14, 2017; date of current version August 25, 2017. This work was supported by Fundamental Research Funds for Central Universities under Grant 3132016331 and Grant 3132017080. *(Corresponding author: Yulei Wang.)*

C.-I. Chang is with the Information and Technology College, Dalian Maritime University, Dalian, China (e-mail: cchang@umbc.edu).

Y. Li is with the Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, Baltimore, MD 21250 USA. (e-mail: liyao1@gmail.com).

Y. Wang is with the Center for Hyperspectral Imaging in Remote Sensing, Information and Technology College, Dalian Maritime University, Dalian, China, and the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: wangyulei.wyl@163.com).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2017.2710219

Recently, a new concept of progressive band processing (PBP) was introduced to process hyperspectral imaging algorithms band by band progressively without waiting for full bands being completely acquired [10]–[14]. Its idea is derived from a data acquisition format, band sequential (BSQ) [15] where bands are acquired band by band. By taking advantage of such interband information provided by PBP, this letter develops a PBP version of FIPPI, to be called PBP-FIPPI which can generate progressive profiles of PPI counts produced by FIPPI band by band to dictate changes of PPI counts between bands. However, this is easier said than done because there is one additional parameter, skewers involved in PBP-FIPPI, which is absent in [10]–[14]. This is because PPI counts always vary with skewers and requires an extra dimensionality to keep track of changes in skewers. It is this parameter of "skewers" that makes PBP-FIPPI quite different from the PBP-versions developed in [10]–[14] because PBP-FIPPI must deal with three parameters, sample vectors, bands, and skewers compared to the PBP-based algorithms in [10]–[14] which only have to work with two parameters, sample vectors and bands. Accordingly, the approaches proposed in [10]–[14] cannot be directly applied to PBP-FIPPI. Most interestingly, PBP-FIPPI can be used to derive a new version of FIPPI, called progressive skewer processing of FIPPI (PSP-FIPPI) which has no counterparts in [10]–[14] because PSP-FIPPI implements FIPPI skewers by skewers progressively. Both PBP-FIPPI and PSP-FIPPI use two loops (inner and outer loops) to carry out iterative processes. Specifically, PBP-FIPPI iterates skewers in an inner loop while iterating bands in an outer loop. By contrast, PSP-FIPPI iterates bands in an inner loop and skewers in an outer loop in which case the number of skewers grows iteration by iteration. So, there are two major differences between PBP-FIPPI and PSP-FIPPI. One is that both implement inner and outer loops reversely. The other is that the number of skewers n_{skewer} must be fixed during data processing by PBP-FIPPI. But the benefit is that PBP-FIPPI can be implemented as a real-time processing algorithm. On the other hand, despite that PSP-FIPPI cannot be implemented in real time as PBP-FIPPI does its great benefit is no need of fixing the number of skewers n_{skewer} which is a significant advantage since in many cases determining an appropriate value of K is very difficult. Nevertheless, both PBP-FIPPI and PS-FIPPI provide significant insights into the design of PPI which cannot be offered by the works in [2]–[6], specifically progressive pro-

1545-598X © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: DALIAN MARITIME UNIVERSITY. Downloaded on December 12,2023 at 13:13:42 UTC from IEEE Xplore. Restrictions apply.

files of interband changes in PPI counts as bands are processed progressively.

II. Progressive Band Processing of FIPPI

FIPPI was developed to resolve two main issues arising in PPI, random initial condition and determination of number of skewers n_{skewer}. FIPPI uses ATGP to generate a specific set of initial skewers as endmembers to avoid randomness caused by skewers and then grows skewers from ATGP-generated initial endmembers instead of fixing the value of n_{skewer}. For the step-by-step implementation of FIPPI we refer its details in [3].

As pointed out in the introduction, the major difficulty in developing PBP-FIPPI is to deal with the parameter of skewers, an issue that does not exist in [10]–[14]. In this section, we extend FIPPI to PBP-FIPPI and PSP-FIPPI, both of which use two loops specified by skewers and bands to process FIPPI depending upon which one is implemented in the outer loop.

A. PBP-FIPPI

The first algorithm iterates the number of skewers, denoted by n_{skewer}, in the inner loop indexed by parameter k, while iterating the number of the first l bands to be used to process the data, denoted by n_l in the outer loop indexed by the parameter l. The resulting algorithm is called PBP-FIPPI.

B. PSP-FIPPI

The second algorithm swaps the inner loop and outer loop implemented in PBP-FIPPI by iterating the number of the first l bands being used, denoted by n_l in the inner loop indexed by the parameter l, while iterating the number of skewers, denoted by n_{skewer}, in the outer loop indexed by parameter k. Since the outer loop produces skewer sets progressively the resulting algorithm is called progressive skewer processing of FIPPI (PSP-FIPPI) described below.

Algorithm 1 Progressive Band Processing of FIPPI (PBP-FIPPI)

Initial Conditions: Find the value of VD, n_{VD} and let it be p.

Outer loop from $l = p$ to L

Inner loop

a. Initial Condition: Let $\left\{\mathbf{skewer}_{l,j}^{(0)}\right\}_{j=1}^{p}$ be an initial set of p skewers generated by ATGP.

b. Normalize $\left\{\mathbf{skewer}_{l,j}^{(0)}\right\}_{j=1}^{p}$ into a unit vector.

c. At iteration k, for each $\mathbf{skewer}_{l,j}^{(k)}$, project all the sample vectors $\{\mathbf{r}_i\}_{i=1}^{N}$ onto the $\mathbf{skewer}_{l,j}^{(k)}$ to find sample vectors which are at its extreme positions to form an extrema set, denoted by $S_{extrema}(\mathbf{skewer}_{l,j}^{(k)})$. Find the sample vector $\mathbf{r}_{l,j}^{(k)}$ that produces the largest $N_{\text{PPI}}(\mathbf{r}_{l,j}^{(k)})$ where $N_{\text{PPI}}(\mathbf{r})$ is defined as as the PPI count of the sample vector $\mathbf{r}$ by

$$N_{\text{PPI}}(\mathbf{r}) = \sum\nolimits_j I_{S_{extrema}(\mathbf{skewer}_{l,j}^{(k)})}(\mathbf{r}) \tag{1}$$

where $S_{extrema}\left(\mathbf{skewer}_{l,j}^{(k)}\right)$ is an extreme set obtained by an indicator function of a set S, $I_S(r)$ defined by

$$I_S(r) = \begin{cases} 1; & \text{if } \mathbf{r} \in S \\ 0; & \text{if } \mathbf{r} \notin S \end{cases}. \tag{2}$$

d. Form the joint set $\left\{\mathbf{skewer}_{l,j}^{(k+1)}\right\} = \left\{\mathbf{r}_{l,j}^{(k)}\right\}_{N_{\text{PPI}}(\mathbf{r}_{l,j}^{(k)})>0} \cup \left\{\mathbf{skewer}_{l,j}^{(k)}\right\}$.

e. If $\left\{\mathbf{skewer}_{l,j}^{(k+1)}\right\} = \left\{\mathbf{skewer}_{l,j}^{(k)}\right\}$, then no new endmembers are added to the skewer set. In this case, the algorithm is terminated, break the inner loop. Otherwise, let $k \leftarrow k+1$, and go to step b.

End Inner loop

End Outer loop

C. Discussions

PBP has been explored for various applications, for example, inverting signature matrices in [10], [12], and [13] and inverting the sample correlation matrix in [11] and [14]. Developing PBP versions of FIPPI was first investigated in [16] where PBP was implemented to account for PPI counts calculated from skewers which are absent in [10]–[14]. Comparing to PBP versions in [10]–[14] which are functions of two parameters, sample vectors and bands, PBP versions of FIPPI are functions of three parameters, sample vectors, bands, and skewers to produce PPI counts which can technically be viewed in a 4-D space. So, there are no corresponding counterparts in [10]–[14]. However, for each given sample vector PBP-FIPPI and PSP-FIPPI can be reduced to functions of two parameters, skewers and bands. In other words, by fixing sample vectors while growing skewers and bands PBP-FIPPI and PSP-FIPPI can be used to see a progressive profile of interband changes in PPI counts for a particular sample vector of interest, such as an endmember. To the authors' best knowledge no such work ever reported in the literature enables to do so.

There are some significant differences between PBP-FIPPI and PSP-FIPPI. In the inner loop, PBP-FIPPI fixes n_l while growing skewer sets as opposed to PSP-FIPPI which fixes a given set of skewers but grows n_l. So, the inner loop of PBP-FIPPI can be considered to perform FIPPI for a given value of n_l. Consequently, the final growing skewer sets produced by different values of n_l in the outer loop of PBP-FIPPI are also different. When $n_l = L$ the skewer set produced by PBP-FIPPI is basically the same skewer set produced by FIPPI. On the other hand, in the inner loop PSP-FIPPI fixes a set of skewers and performs PPI as does the inner loop of PBP-FIPPI in which case the inner loop of PSP-FIPPI can be considered as PBP-PPI as n_l is increased. More importantly, PBP-FIPPI grows skewer sets in the inner loop, whereas PSP-FIPPI grows skewer sets in the outer loop. Therefore, their results are different.

Authorized licensed use limited to: DALIAN MARITIME UNIVERSITY. Downloaded on December 12,2023 at 13:13:42 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2 Progressive-Skewer-Processing of FIPPI (PSP-FIPPI)

Initial Conditions:

1. Find the value of VD, n_{VD} and let it be p.
2. Let $\left\{\mathbf{skewer}_{l,j}^{(0)}\right\}_{j=1}^{p}$ be an initial set of p skewers generated by ATGP.
3. Normalize $\left\{\mathbf{skewer}_{l,j}^{(0)}\right\}_{j=1}^{p}$ into a unit vector.

Outer loop indexed by k

Inner loop from $l = p$ to L

a. Input band l

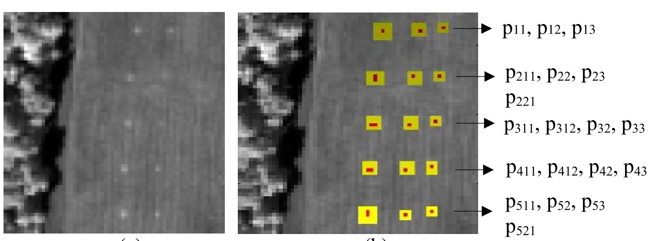
b. At iteration k, for each $\mathbf{skewer}_{l,j}^{(k)}$, project all the sample vectors $\{\mathbf{r}_i\}_{i=1}^{N}$ onto this particular $\mathbf{skewer}_{l,j}^{(k)}$ to find those which are at its extreme positions to form an extrema set, denoted by $S_{extrema}(\mathbf{skewer}_{l,j}^{(k)})$. Find the sample vectors that produce the largest $N_{PPI}(\mathbf{r}_{l,j}^{(k)})$ and let them be denoted by $\left\{\mathbf{r}_{l,j}^{(k)}\right\}$.

c. Form the joint set $\left\{\mathbf{skewer}_{l,j}^{(k+1)}\right\} = \left\{\mathbf{r}_{l,j}^{(k)}\right\}_{N_{PPI}(r_{l,j}^{(k)})>0} \cup \left\{\mathbf{skewer}_{l,j}^{(k)}\right\}$.

End Inner loop

If $\left\{\mathbf{skewer}_{l,j}^{(k+1)}\right\} = \left\{\mathbf{skewer}_{l,j}^{(k)}\right\}$, then no new endmembers are added to the skewer set. In this case, the algorithm is terminated, exit the outer loop. Otherwise, let $k \leftarrow k+1$.

End Outer loop



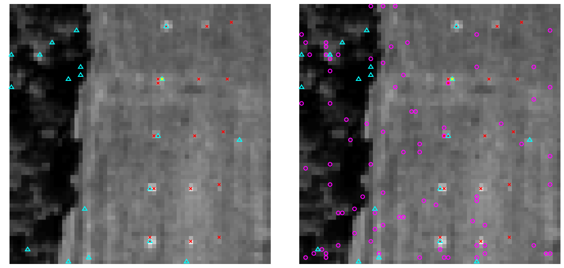
(a) (b)

Fig. 1. (a) HYDICE panel scene which contains 15 panels. (b) Ground truth map of spatial locations of the 15 panels.

III. Real Image Experiments

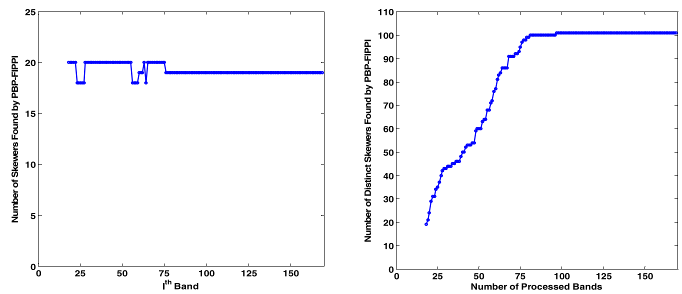
A real hyperspectral image scene collected by HYperspectral Digital Imagery Collection Experiments (HYDICE) is shown in Fig. 1(a), which has been studied extensively. More details about this scene can be found in [2]. It was used for experiments to demonstrate the utility of the PBP-FIPPI and PSP-FIPPI in finding endmembers.

Fig. 1(b) provides the ground truth of Fig. 1(a) where there are 15 panels of three different sizes, 3 m $\times$ 3 m and 2 m $\times$ 2 m and 1 m $\times$ 1 m with 19 center pixels highlighted by RED, three panel pixels p_{11}, p_{12}, and p_{13} in row 1, four panel pixels p_{211}, p_{221}, p_{22}, and p_{23} in row 2, four panel pixels p_{211}, p_{312}, p_{32}, and p_{33} in row 3, four panel pixels p_{411}, p_{412}, p_{42}, and p_{43} in row 4, four panel pixels p_{511}, p_{521}, p_{52}, and p_{53} in row 5. Also, the pixels in yellow (Y pixels) in Fig. 1(b) are panel pixels mixed with the background. According to the ground truth the panel pixels are true endmembers since these panels are made by a single material fabric which can be considered as pure signatures.



(a) (b)

Fig. 2. Comparison of endmember candidates extracted between FIPPI and PBP-FIPPI using $n_{\mathrm{VD}} = 18$. (a) FIPPI. (b) PBP-FIPPI.



(a) (b)

Fig. 3. Result of PBP-FIPPI. (a) Number of skewers found by PBP-FIPPI in each band. (b) Number of distinct endmember candidates identified by PBP-FIPPI versus number of processed bands.

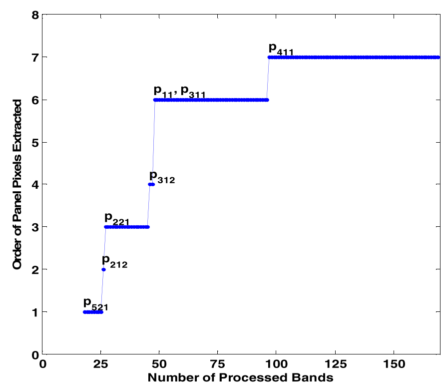


Fig. 4. Orders of extracted panel pixels versus n_l.

A. PBP-FIPPI

Fig. 2(a) and (b) shows endmember candidates extracted from the HYDICE scene by FIPPI and PBP-FIPPI using $n_{\mathrm{VD}} = 18$ [17], respectively, where FIPPI extracted 4 R-panel pixels, p_{11}, p_{311}, p_{411}, and p_{521} plus 1 Y-panel pixel, p_{212} as opposed to 6 R-panel pixels, p_{11}, p_{221}, p_{311}, p_{312}, p_{411}, and p_{521} plus 1 Y-panel pixel, p_{212} extracted by PBP-FIPPI where the R-panel pixels, the Y-panel pixels, the endmember candidates extracted by previous bands, and the endmember candidates extracted by the current band are highlighted by red crosses, yellow crosses, the magenta circles and the cyan upper triangles, respectively. It also shows that PBP-FIPPI produced the same results as FIPPI did when it reached the last band. In particular, many target pixels marked by magenta circles in Fig. 2(b) were extracted by PBP-FIPPI as additional endmember candidates during band-by-band processing which were not picked up by FIPPI. Specifically, PBP-FIPPI extracted 2 additional R-panel pixels p_{221} and p_{312} which were missed by FIPPI.

Authorized licensed use limited to: DALIAN MARITIME UNIVERSITY. Downloaded on December 12,2023 at 13:13:42 UTC from IEEE Xplore. Restrictions apply.

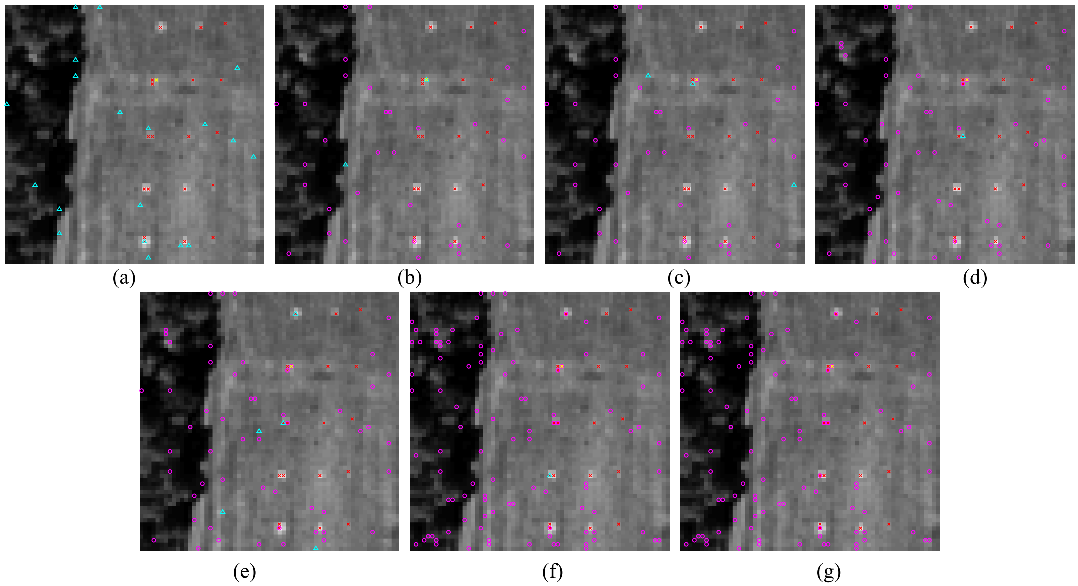


Fig. 5. Endmember candidates identified by PBP-FIPPI with different number of bands processed, n_l. (a) 18 bands. (b) 26 bands. (c) 27 bands. (d) 46 bands. (e) 48 bands. (f) 97 bands. (g) 169 bands (full bands).

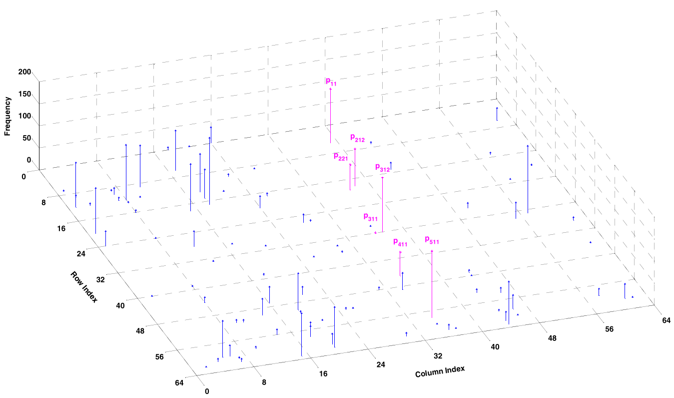


Fig. 6. Three-dimensional histogram of the pixels picked up by PBP-FIPPI as endmember candidates after processing full 169 bands.

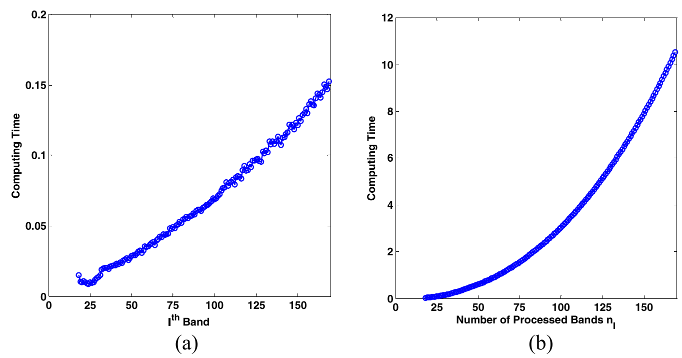


Fig. 7. Computing time versus lth band and n_l required by PBP-FIPPI. (a) lth band. (b) n_l-bands.

Furthermore, Fig. 3(a) also plots the number of endmember candidates extracted by each band and Fig. 3(b) shows the accumulative distinct endmember candidates extracted by PBP-FIPPI as the number of processed the first l bands, n_l is increased. As shown in Fig. 3(b), a total of 101 distinct target pixels were extracted as endmember candidates. As $n_l \geq 97$ no more endmember candidates were extracted.

Fig. 4 plots the order of these panel pixels being extracted versus n_l where it clearly shows that, p_{521} was the first R-panel pixel being extracted using only $n_l = 18$ bands. After $n_l \geq 97$ in Fig. 4, all the 7 R-panel pixels were eventually extracted by PBP-FIPPI as endmember candidates.

Fig. 5(a)–(g) further shows the spatial locations of the endmember candidates identified by PBP-FIPPI using different number of bands processed with n_l starting from the first 18 bands. In Fig. 5(a)–(g), the spatial locations of the ground truth pixels, the endmember candidates extracted by the previous bands and the endmember candidates by the current band are highlighted by red crosses, yellow circles, and the cyan upper triangles, respectively. When PBP-FIPPI began with $n_l = 18$, the R-panel pixel p_{521} was immediately extracted. It was followed by p_{212} with $n_l = 26$ bands being processed. After $n_l \geq 97$ bands, all the 6 R-panel pixels were eventually extracted by PBP-FIPPI and no more panel pixels are extracted.

Moreover, Fig. 6 demonstrates a 3-D plot to show how frequently a particular pixel is picked up by PBP-FIPPI as an endmember candidate where the magenta arrows indicate the panel pixels; the x-axis and y-axis correspond to the columns and rows of the HYDICE scene and the z-axis indicates the number of times a given pixel was detected as an endmember pixel by PBP-FIPPI after processing all bands. Such progressive profiles provided by FIPPI in Fig. 6 are unique advantages and benefits that cannot be offered by any PPI-based method.

Finally, Fig. 7(a) plots computing time versus the lth band where PBP-FIPPI was executed in MATLAB R2012B with an Intel Core i7-3770 running at 3.40 GHz with 16 GB of RAM. The results were obtained by running PBP-FIPPI ten times to produce an average computing time. When the lth band reached the last band, 169 the computing time was 0.1524 as opposed to FIPPI which required 0.1756 s using full 169 bands.

Authorized licensed use limited to: DALIAN MARITIME UNIVERSITY. Downloaded on December 12,2023 at 13:13:42 UTC from IEEE Xplore. Restrictions apply.

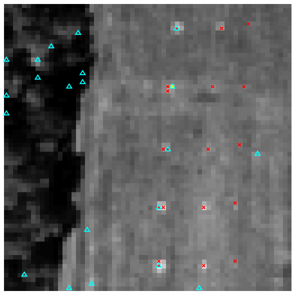
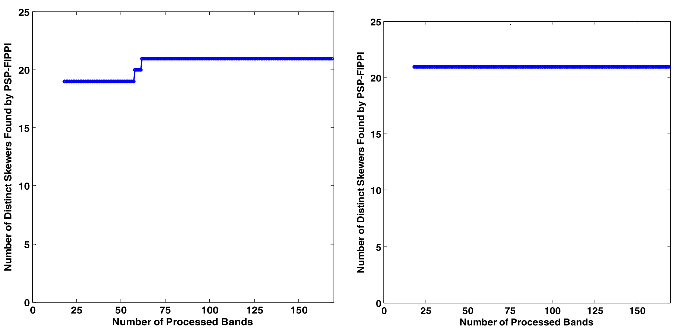


Fig. 8. PSP-FIPPI results using $n_{\mathrm{VD}} = 18$.



(a) (b)

Fig. 9. Plots of number of distinct endmember candidates extracted by PSP-FIPPI versus number of bands processed, n_l in each iteration. (a) First iteration. (b) Second iteration.

Fig. 7(b) also plots the accumulative computing time versus n_l required by PBP-FIPPI where approximate 10.5 s were required to process all bands. However, the payoff is that PBP-FIPPI provides 152 progressive profiles of pixels as the first band numbers n_l increased from 18 to 169.

B. B. PSP-FIPPI

Fig. 8 plots the results of applying PSP-FIPPI to the HYDICE scene using $n_{\mathrm{VD}} = 18$.

As shown in Fig. 8, the spatial locations of the ground truth pixels and the endmember candidates extracted by the current band are highlighted by red crosses and the cyan upper triangles, respectively, where 4 R-panel pixels and 1 Y-panel pixel were extracted as endmember candidates by PSP-FIPPI.

Furthermore, PSP-FIPPI required only two iterations to terminate the outer loop. Fig. 9(a) plots the number of distinct endmember candidates extracted by PSP-FIPPI versus n_l in each outer iterations. As shown in Fig. 9(b), after the first outer iteration, there were 21 endmember candidates extracted by PSP-FIPPI and then no additional pixels were extracted as endmembers afterwards. This is the reason that the plot in Fig. 7(b) is flat.

IV. CONCLUSION

FIPPI was previously developed to address two major issues encountered in PPI: 1) the use of skewers whose number must be determined *a priori* and 2) inconsistent final results which cannot be re-produced. Recently, a new concept of PBP has been developed for hyperspectral data communication according to the BSQ acquisition format in such a way that data can be processed while band-by-band acquisition is ongoing. Its advantages have been justified in several applications such as anomaly detection [14], constrained energy minimization [11], ATGP [12], and orthogonal subspace projection [10]. This letter further extends FIPPI to two progressive versions of FIPPI, PBP-FIPPI, and PSP-FIPPI by growing skewer sets as well as first band numbers n_l via two iterative loops. When the outer loop is iterated by growing n_l, it is PBP-FIPPI. When the outer loop is iterated by growing skewers, it is PSP-FIPPI. Interestingly, both versions provide different insights into the design of FIPPI but produce slightly different results. More details regarding PBP-FIPPI and PSP-FIPPI such as their synthetic image experiments and GUI design can be found in [16].

REFERENCES

[1] J. W. Boardman, "Geometric mixture analysis of imaging spectrometry data," in *Proc. Int. Geosci. Remote Sens. Symp.*, vol. 4. Aug. 1994, pp. 2369–2371.

[2] C.-I. Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*. Hoboken, NJ, USA: Wiley, 2013, chs. 7–11.

[3] C.-I. Chang and A. Plaza, "A fast iterative algorithm for implementation of pixel purity index," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 63–67, Jan. 2006.

[4] F. Chaudhry, C. Wu, W. Liu, C.-I Chang, and A. Plaza, "Pixel purity index-based algorithms for endemember extraction from hyperspectral imagery," in *Recent Advances in Hyperspectral Signal and Image Processing*, C.-I Chang, Ed. Trivandrum, Kerala: Research Signpost, 2006, ch. 2.

[5] C. I. Chang, C. C. Wu, and H. M. Chen, "Random pixel purity index," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 2, pp. 324–328, Apr. 2010.

[6] C.-I. Chang and C. C. Wu, "Design and development of iterative pixel purity index," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 7, pp. 2583–2597, Jul. 2015.

[7] C.-I. Chang, *Hyperspectral Imaging: Spectral Techniques for Detection and Classification*. Norwell, MA, USA: Kluwer, 2003.

[8] C.-I. Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 3, pp. 608–619, Mar. 2004.

[9] H. Ren and C.-I. Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1232–1249, Oct. 2003.

[10] H. C. Li, C.-I. Chang, and M. Song, "Progressive band processing of orthogonal subspace projection," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 3–7, Jan. 2016.

[11] C.-I. Chang, R. Schultz, M. Hobbs, S.-Y. Chen, Y. Wang, and C. Liu, "Progressive band processing of constrained energy minimization for subpixel detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1626–1637, Mar. 2015.

[12] C.-I. Chang and Y. Li, "Recursive band processing of automatic target generation process for subpixel detection in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5081–5094, Sep. 2016.

[13] C.-I. Chang, C. C. Wu, K. H. Liu, H. M. Chen, C. C. C. Chen, and C. H. Wen, "Progressive band processing of linear spectral unmixing for hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2583–2597, Jun. 2015.

[14] C.-I. Chang, Y. Li, M. Hobbs, R. Schultz, and W. M. Liu, "Progressive band processing of anomaly detection in hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 7, pp. 3558–3571, Jul. 2015.

[15] R. G. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, 2nd ed. San Diego, CA, USA: Academic, 1997.

[16] Y. Li, "Recursive band processing algorithms for finding unknown targets in hyperspectral imagery," Ph.D. dissertation, Dept. Comput. Sci. Elect. Eng., Univ. Maryland, Baltimore, MD, USA, Dec. 2016.

[17] C.-I. Chang, X. Jiao, C.-C. Wu, E. Y. Du, and H.-M. Chen, "Component analysis-based unsupervised linear spectral mixture analysis for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4123–4137, Nov. 2011.

Authorized licensed use limited to: DALIAN MARITIME UNIVERSITY. Downloaded on December 12,2023 at 13:13:42 UTC from IEEE Xplore. Restrictions apply.