Chapter 3

Deep CNN Feature Fusion with Manifold Learning and Regression for semantic pixel classification

As discussed in objective-1, this chapter studies the role of CNN based feature extraction method followed by a robust manifold learning and a regression-based classification framework for pixel-wise semantic scene labelling.

3.1 Introduction

Supervised classification and target recognition of Hyperspectral images(HSI) is a challenging task due to high dimensionality and spectral-mixing in pixels. Straightforward cognitive computation and target classification lead to high computation cost and low recognition accuracy. Limited availability of training samples makes the recognition process very slow and inaccurate. The main purpose of this work is to improve the classification accuracy for high dimensional images by the fusion of posterior probability obtained from the two-stage probabilistic framework. The first stage addresses the issue of high dimensionality, and the second stage addresses the spectral mixing problem. Both stages provide the prediction probability of pixels in a particular class. In stage-1, we have addressed the imbalance between dimensionality and training samples problem for which

we have integrated the deep CNN based spatial and spectral features in combined data-cube form, using 'off-the-shelf' CNN models. Subsequently, a graph-based non-linear manifold embedding has performed to extract and fuse the region-wise external information. A probability of prediction has obtained by using the LDA classifier. These probabilistic values have denoted as a global probability, as an outcome of stage-1. In stage-2, the spectral mixing issue has addressed by computing the regional probabilities of class mixing for each pixel. The regional probabilities have calculated by using a regional subspace regression approach. Subsequently, the probabilistic output, obtained from stage-1 and stage-2, has been combined with a linear decision fusion method using regularizers. The experiments have conducted on three real Hyperspectral images .i.e. Indian pines(IP), Pavia University(PU), Salinas Valley(SV) datasets. The probabilistic fusion of stage-1 and stage-2 yields to the maximum overall accuracy of 97.38%, 95.10%, and 99.88% for IP, PU, and SV datasets. The over-all accuracies have compared with past methods, and it has found that the proposed framework is providing higher prediction accuracies than previous state-of-art methods.

3.2 Proposed work

Figure-3.1(a) and (b) Describes the structure of our proposed scheme, which exhibits two sections:1)DeepCNN based feature extraction and 2) The concatenation of spatial and spectral features followed by a fusion of regional and global probabilities of pixels. In section-3.2.1, we have described the deep feature extraction using the convolution layers of the AlexNet model. In section-3.2.2, the global probability estimation method has discussed. Section-3.2.3 has depicted the process of regional probability calculation and elaborates on the process for the generation of local class mixing images. Section-3.2.3.1 has discussed the subspace projection and regression to obtain regional probability. Finally, the section-3.2.3.3 has explained the linear fusion of global and regional probabilities using regularizers.

3.2.1 Deep CNN Spatial Feature by off the self networks

In forward propagation of Deep architecture of CNNs, the features in the previous layer

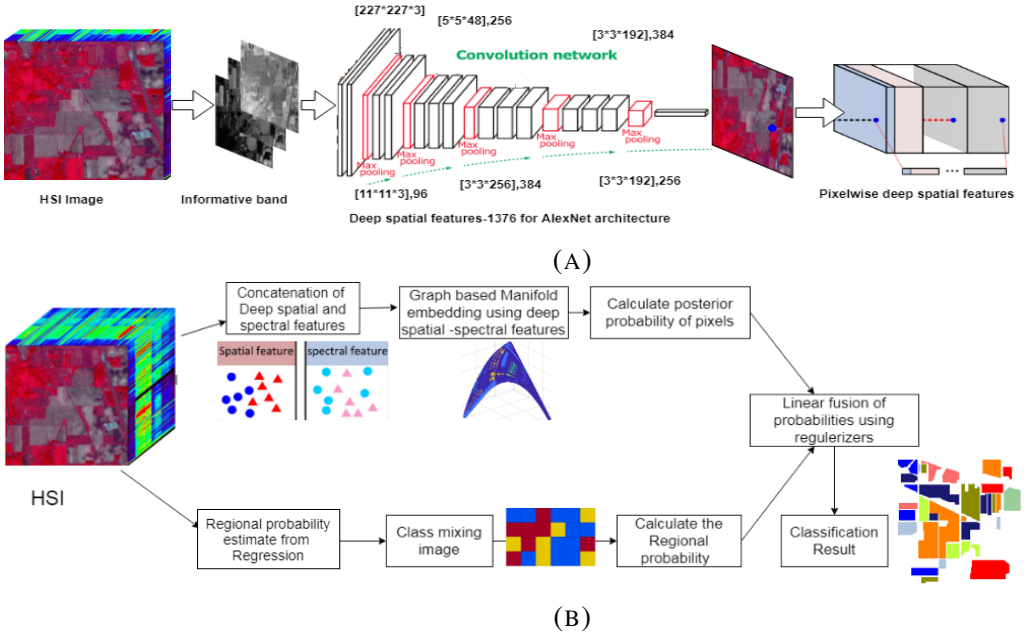


FIGURE 3.1: Deep feature extraction framework using principle components followed by convolution map generation

maintain the higher spatial resolution for contextual information extraction. In contrast, subsequent layers occupy more semantic information and less spatial knowledge. After all, we are interested in capturing the exact spatial detail [42]. Therefore, we have avoided the fully connected layer. The deep feature extraction process has inspired from the [49] in which a fully-connected layer exhibits the small spatial resolution. We have adapted the hierarchical levels of up-sampling on the convolution layer to extract the spatial features by using spatial details and spectral semantics. Up-sampling has performed by applying the activation on convolution layers. We described the deep spatial features for an HSI pixel as an array of activation for all CNN convolution units at that pixel, as shown in figure-3.1(a). Categorically, we have first observed the principal components(pc's) by using all spectral channels for a given image. Subsequently, we extracted their hierarchical convolution level features by 'off-the-shelf' CNN namely [35], VGG-16[35],VGG-19[35] trained on imagenet data for 1000 classes, using 3 pc's as input. Up-sampling and pooling operation reduces the spatial resolution of feature maps as the depth of the convolution layer increases. To point out this problem, we have up-sampled the feature maps by using bi-cubic interpolation. Lastly, up-sampled feature maps have concatenated from each convolution layer in a long array of pixel vector, which we have described as deep spatial features for that position. For Example, convolution layer-1 to 5 in [35], which have 96,

256, 384 and 256 features, will generate 1376 cumulative features for every pixel.

3.2.2 Estimation of Global probability using Manifold learning

In this section, a graph-based embedding framework has applied for global probability estimation. Let our dataset $X = (x_i)_{(i=1)}^{k}$ with d dimensions has projected in a S^f feature space where k is the number of samples and f is projected to feature space dimension. Manifold learning method has adapted a graph embedding framework in which $G = (X, W)$ is a weighted graph. W is a $k \times k$ adjacency matrix, also referred to as an affinity matrix. The algorithm utilizes the idea of affinity weights $W_{i,j}$ to measure the distance between two pixels. The generation and fusion of spectral weights with spatial potential have carried out from [39] in which the affinity function classifies the neighbourhood relation between all pairs of pixels based on their feature distance, hence preserves the local properties. The weight W represents the edge between pixels nodes. shi-mallik has proposed an edge construction method in which differences in spectral values has used. But in HD images, the difference in the direction of values is an essential concern for better feature extraction. Gillies bowels have improved the shi-malik weights by adding a penalty on the changes in the direction of spectral values. Therefore in this work, the Gillies bowels weights have used for the construction of weighted edges as:

$$W_{i,j} = \begin{cases} exp\left(-cos^{-}1\left(\frac{(x_i^f, x_j^f)}{||x_i^f||.||x_j^f||}\right)\right), & (x_i^f, x_j^f)\varepsilon W \\ 0, & otherwise \end{cases} \tag{3.1}$$

The steps for graph construction, manifold embedding and global probability estimation are:

1. Design an adjacency matrix A of graph $G = (X, W)$ in which pixels are vertices and edges are the weighted affinity between the pixels as described in equation-1.
2. Construct a Laplace matrix from a given weighted adjacency matrix as L = D-A. where D is a diagonal matrix of A

3. Construct a spatial potential matrix V as

$$V = \Sigma_{\{i=1\}}^{K} \left(\Sigma_{\{x_j \varepsilon N_p^{(x_i)}\}} V_{(i,j)}^{(k,l)} . \gamma . exp^{-\left(\frac{|x_i^p - x_j^p|^2}{\sigma_p^2}\right)} \right) \tag{3.2}$$

Where

$$V_{i,j}^{(k,l)} = \begin{cases} 1, & (k,l)\varepsilon(i,i),(j,j) \\ -1, & (k,l)\varepsilon(i,j),(j,i) \\ 0, & otherwise \end{cases}$$

and $\gamma = \frac{spectral\ dimesnsion}{spatial\ dimension}$, which provides the greater influence for spatial neighbours. x_i^p and x_j^p are the spatial coordinates of image. $N_p^{(x_i)}$ are the points in k-nearest neighbour proximity of x_i.

4. Combine spatial and spectral potential matrix is constructed as $L + \alpha V$. The Gilles bowles weights are changed as:

$$W_{i,j} = \begin{cases} exp^{\left(-cos^{-}1\left(\frac{(x_i^f, x_j^f)}{||x_i^f||.||x_j^f||}\right)\right) - \left(\frac{|x_i^p - x_j^p|^2}{\sigma_p^2}\right)}, & (x_i^f, x_j^f)\varepsilon W \\ 0, & otherwise \end{cases} \tag{3.3}$$

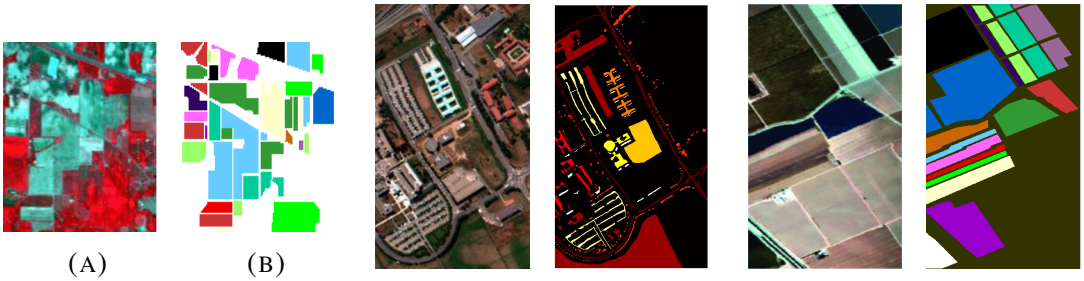
5. The Laplace eigenvalue generalization equation is changed as:

$$(L + \alpha V) f = \lambda . D . f \tag{3.4}$$

6. Calculate the k+1 smallest eigenvectors of equation-4. D is diagonal of weights i.e. $D_{i,i} = \sum_j W_{i,j}$ and L=D-W, which is Laplace matrix. The obtained eigenvectors f of our pixels are the new values in transformed manifold. The eigenvectors are in increasing order as $f_0, f_1, .., f_k$. Therefore, transformed pixel values in new manifold are F=[$f_1, f_2, .., f_k$];

7. Further, a Fisher's Linear Discriminant has been applied to obtain global posterior probability of manifold pixel value belongs to ground truth classes $c_1, c_2, ..., c_i$ as:

$$p_g(f_i|c_i) \ \forall \ i \ \varepsilon \ k$$



(A) (B) (C) (D) (E) (F)

FIGURE 3.2: False colour image and Ground truth for (a)Indian Pines(IP)(b)Pavia University(PU) (c)Salinas valley(SV) datasets

3.2.3 Estimation of Regional probability by mixing of classes in pixel using Subspace Projection

The pixels of the High Dimensional image faces a well-known spectral signal mixing problem. In the classification process, the mixed pixels cause a hurdle in classification of a pixel in the right class. Therefore, the mixing of various classes within the pixels has assumed. A mixed class image has been generated to obtain the regional probability of mixed pixels. The mixing of classes in a pixel has contributed by the posterior probability of a subspace multinomial regression(subspaceMLR) method. Therefore, a subgroup I(mixed component) of available classes has selected for each pixel as a class mixing group. Let c is the number of classes, therefore $I \leq c$. For I=c, the regional probability is equal to global probability. We have adapted the process of class mixing images for regional probability estimation from [44], which is a benchmark work in the local probability estimate in a mixed pixel environment. In the referred work, the authors have obtained the regional probabilities, also denoted as local probabilities, by using a weighted prior in MLR regression. Whereas, we have projected the subspace by using Hysime method [43] followed by composing a class mixing image. Subsequently, we have applied logistic regression on the projected-mixed image. For example, in figure-3.3, we have shown the process of generating a mixed pixel map for I=2(mixing classes) in a demo image. In the original image, the number of classes is three and denoted as (A, B, C). An array of posterior probabilities for three classes (A, B, C) has obtained from the previously

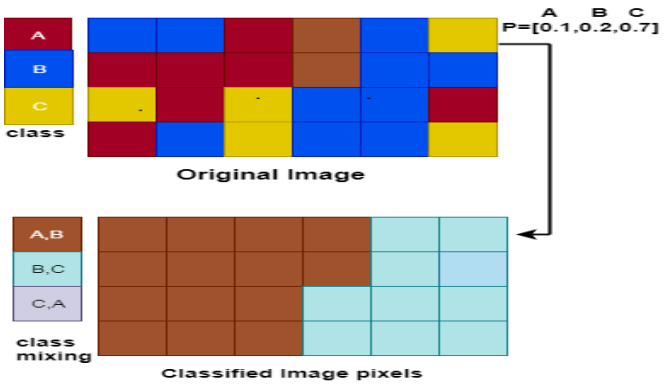


FIGURE 3.3: original image and class mixing image for I=2

mentioned Hysime based subspaceMLR method. Let the upper right pixel has the posterior probabilities as 0.1, 0.2, and 0.7 for respective A, B, and C classes. We have allocated the respective pixel to the highest probability classes .i.e. 0.2 and 0.7. Therefore, The pixel in the mixed image will allocate to the mixed class(B, C). The number of class mixing combinations are $3C_2$ .i.e 3. It is to notice that the mix-class (C, A) is not present in the final class mixing image.

3.2.3.1 Subspace projection based MLR method:subspaceMLR

For a HSI dataset, $H_i = \{x_{i1}, x_{i2}..., x_{in}\}$, $(x_{i1},\ x_{i2}, .., x_{in})$ denotes the pixels vector where $i \varepsilon C$ and C=[1,2,3,...,n] for n pixels of an image. Our objective in pixel-wise probabilistic classifier is to calculate a decision probability of individual pixel for a set of assigned labels $y_i \varepsilon (1,2,..,K)$, where K denotes the number of classes. The decision probability is calculated by a posterior estimation of probability for each pixel belonging to K classes .i.e. $p(y_i = K|x_i)$. Therefore, our problem is converted into a MAP(maximum a posteriori) based estimate, as described in problem statement section. Discriminant function is defined as: $y_i = k_1$ if $p(y_i = k_1|x_i) > p(y_i = k_2|x_i)$, where $(k1, k2\ \varepsilon\ K)$. Numerous algorithms such as [41], are proposed to solve this MAP based estimate. In this paper, we have used HySime based subspace projection method, proposed in the reference [43]. By using HySime method, we have obtained the projected dimensions with un-mixed pixels. Subsequently, we have applied the logistic regression to obtain the regional probability of pixels. By this method, we have modelled our MAP equation directly to posterior estimate of pixels probability for each class label. In this paper, the discriminative and generative probability,

.i.e $p(x_i|y_i)$ and $p(y_i)$, is resulting in MAP based class prediction .i.e $p(y_i|x_i)$:

$$\hat{p}(y_i = k|x_i) = \frac{exp(\phi_{(k)}(x_i))}{\Sigma_{k=1}^{K} exp(\phi_{(k)}(x_i))} \tag{3.5}$$

The subspace projection idea has based on a fundamental assumption that any high dimensional dataset can reside in lower space without loss of information. Therefore, classes have represented as orthonormal basis vectors. Hence the feature vector function $\phi_{(k)}(x_i)$ is denoted as:

$$\phi_{(k)}(x_i) = [||x_i^2||, ||x_i^T U_{(k)}||^2]$$

where $U_{(k)}$ is the feature vector of length k. This method is applied in this paper with the annotation as subspaceMLR, for posterior probability estimation.

3.2.3.2 Regional probability calculation

In this section, we have described the regional probability estimation method, which deals with local noise and spectral-mixing in HSI,s. In [43], it has found that HySime method is efficient enough to deal with spectral mixing in HSIs. Therefore, HySime and MLR(Multinomial Logistic Regression) methods are applied to estimate the regional probabilities. The regional posterior probability for I mixed classes has calculated as described in the previous section. The regional probability of pixels has denoted by p_r. For example, in figure-3, the regional probability of uppermost right pixel is:

$$p_{rk} = [0, p(y_{kB} = 1|x_i), p(y_{kC} = 1|x_i)] \tag{3.6}$$

where p_{rk} is local probability of k^{th} pixel. It is to notice that the subspaceMLR method is regionally learning the posterior probabilities from class mixing images. In the regional probability calculation, only I classes have used for mixing in each pixel, and the remaining classes have set to zero. For example, in equation-3.6, $p(y_{kA} = 0)$ implies that the probability of class A in an upper right pixel is irrelevant for spectral mixing. Therefore, we have set it to 0 in class mixing image. In this way, irrelevant classes have removed for each pixel regionally. It has observed that the removal of less relevant classes in class mixing image leads to reduce local noise in the image and hence improved the pixel classification.

3.2.3.3 Fusion of regional and global Probabilities using Regularizer

The regional and global probabilities have been integrated to produce a final probability as described in linear consensus-based probabilistic fusion [33]. Two sources of posterior probabilities have integrated with a regularizer λ_1 and λ_2. λ_1 and λ_2 denotes the relative weight of regional and global posterior estimates.

$$p(y_{kC}|x_k) = \lambda_1 p_g(y_{kC}|x_k) + \lambda_2 p_r(y_{kC}|x_k) \tag{3.7}$$

s.t $\lambda_1 + \lambda_2$=1 and $0 \leq \lambda_1 \leq 1, 0 \leq \lambda_2 \leq 1$. In the above equation, if λ_1=0, then only regional probability has considered for the final estimate, and if λ_2=0, then global probability will be the final estimate. The impact of regularizer, for local posterior, λ_2 on overall accuracy, has been discussed in the experimental section.

3.3 Experimental Result Analysis

3.3.1 Dataset detail

The proposed method has evaluated on three publicly available high dimensional images (HDI) by using standard metrics such as overall accuracy(OA), average accuracy(AA). The OA is calculated by dividing the number of correctly predicted test samples by the total number of test samples, i.e.($OA = \frac{Number\ of\ correctly\ predicted\ test\ samples}{Total\ number\ of\ test\ samples}$). AA is the average of each accuracy per class, i.e.($AA = \frac{Sum\ of\ correct\ predictions\ in\ each\ class}{Number\ of\ classes}$). Indian pines(IP) image has a size of $145 \times 145 \times 200$ in 0.4 to 2.5 μ meter range with 16 classes in the ground truth image. Pavia University(PU) Dataset contains $610 \times 340 \times 103$ pixels in 0.43 to 0.86 μ meters range. Ground truth of PU contains nine exclusive classes. The Salinas Valley(SV) image contains the $512 \times 217 \times 204$ pixels with 16 classes in ground truth. Figure-3.2(a),(c),(e) shows the false colour image and Figure-3.2(b),(d),(f) shows the ground truth for Indian pins(IP), Pavia University(PU) and Salinas Valley (SV) datasets respectively.

TABLE 3.1: Spatial features extracted for different architecture

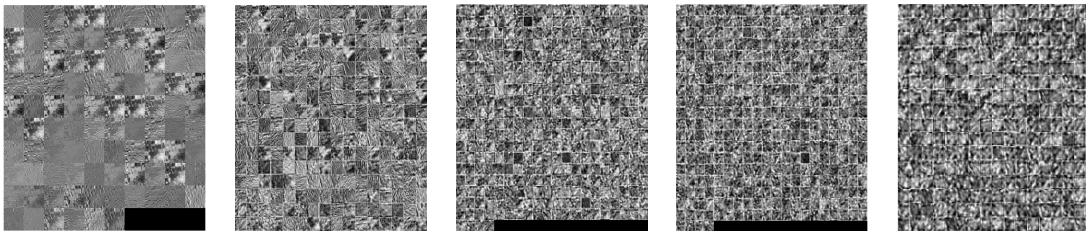
CNN	Layers	Features(IP)	Features(Pavia)	Features(Salinas)
AlexNet[1]	conv1-conv5	1376	1372	1380
VGG-16[2]	conv1-conv5	1472	1372	1476
VGG-19[2,3]	conv1-conv5	1472	1472	1476

3.3.2 Deep spatial features extraction

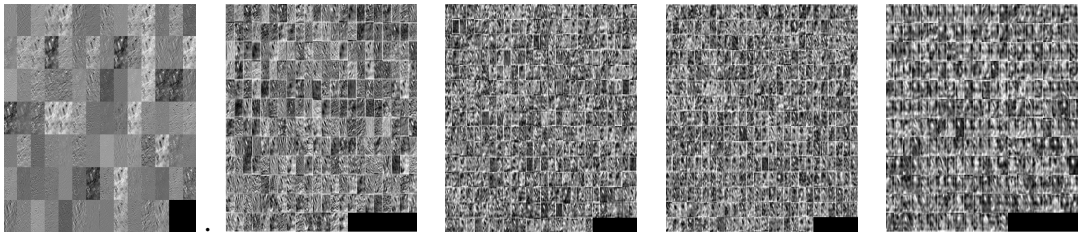
For Deep-CNN learning, the architecture of our network has modified in such a way that all the three 'off the shelf' CNN's learns for three given HD images (HDI). To do so, a new, fully connected layer with 'weight learning rate factor' and 'bias learning rate' =20 with a softmax and classification layer for three HDI categories have been replaced in CNNs. The training method is 'stochastic gradient descent with momentum(sgdm)', learning rate schedule is 'piecewise', drop factor is 0.2, drop period is 5, maximum Epochs are 10, and mini-batch size is 64. The layer activation function, followed by a 'bicubic' interpolation, has applied for the up-sampling of features. In this experiment, we focussed on the working of three CNN models, namely Alexnet [42], VGG16 [34] and VGG19 [34], trained on ImageNet image set. Originally, the models have trained for 1000 classes on ImageNet. Therefore, We have changed the number of class categories to 3 and updated the last layer using transfer learning. Table-3.1 shows the number of deep spatial features extracted by above mentioned 'off-the-shelf' CNN architectures. It is to note that, in VGG-16 and 19, we have considered the features of the last layer in case of more than one convolution layer at a single point layer. Deep spatial features from different convolution layer of AlexNet architectures have shown in figure-3.4. Likewise, Deep spatial features for VGG-16 and 19 have extracted. The Deep features are concatenated with spectral features to obtain a combined spatial-spectral datacube image for IP, PU, and SV. The concatenation procedure has described in figure 3.1(a). In subsequent sections, Global and regional probabilities have calculated for these combined datacube images.

3.3.3 Global probability estimate with manifold learning

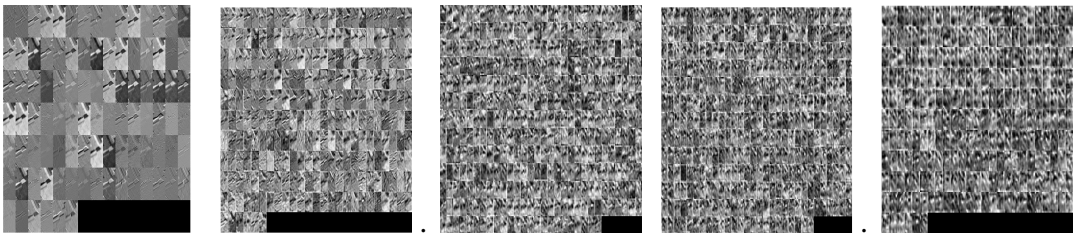
In this section, a graph-based manifold embedding is performed for spatial and spectral data integration, as described in section-3.2.2. In figure-3.5(a), (b), (c), the data distribution of spectral features (original image) has shown. Figure-3.5(d), (e), (f) shows



(A) conv1(96)IP (B) conv2(256)IP (C) conv3(384)IP (D) conv4(384)IP (E) conv5(256)IP

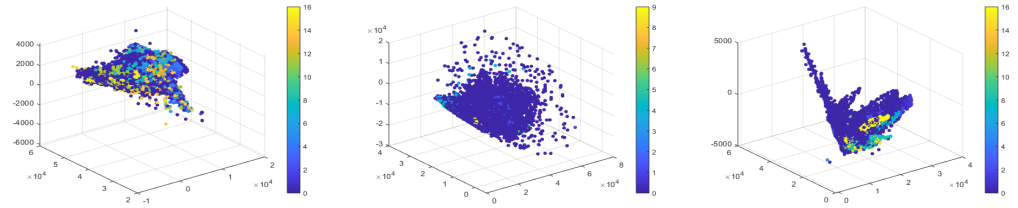


(F) conv1(96)PU (G) conv2(256)PU (H) conv3(384)PU (I) con4(384)PU (J) conv5(256)PU

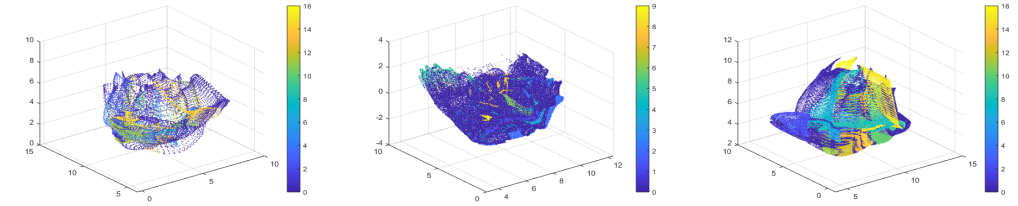


(K) conv1(96)SV (L) conv2(256)SV (M) conv(384)SV (N) conv4(384)SV (O) conv5(256)SV

FIGURE 3.4: Deep spatial features for IP,PU and SV in AlexNet

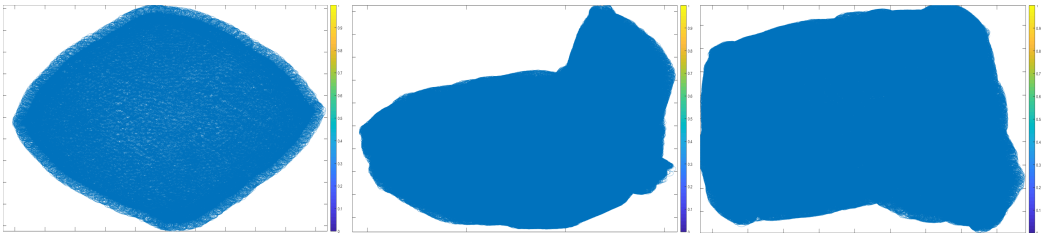


(A) IP spectral pc's (B) PU spectral pc's (C) SV spectral pc's



(D) IP-AlexNet Deep spatial-spectral pc's (E) PU-AlexNet Deep spatial-spectral pc's (F) SV-AlexNet Deep spatial-spectral pc's

FIGURE 3.5: Data distribution on principal components(pc) with spectral datacube(a, b, c) and Deep spatial-spectral datacube(d, e, f)



(A) Laplace graph(L) for IP of AlexNet (B) Laplace graph(L) for PU of AlexNet (C) Laplace graph(L) for SV of AlexNet

(D) Spatial potential graph for IP (E) Spatial potential graph for PU (F) Spatial potential graph for SV

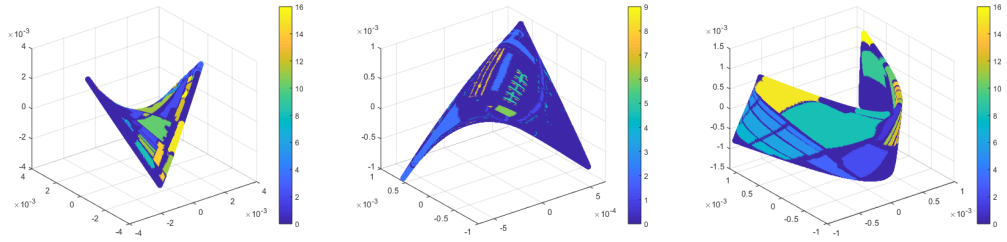
FIGURE 3.6: Laplace and spatial potential graphs of IP, PU,SV for AlexNet architecture

the data distribution of combined spectral and deep-spatial features, obtained from AlexNet architecture. It has been observed that data point in the first three principal components are better separable when deep-spatial features have combined with spectral features. Therefore, we have used a combined deep spatial-spectral datacube for the manifold learning method. The combined datacube has denoted as spatial-spectral datacube for short. Subsequently, we have performed the manifold embedding followed by posterior probability estimation using LDA(Linear Discriminant Analysis). Manifold embedding has performed in two steps. Firstly, a weighted graph of local similarity has been constructed with Gillies Bowels weights as described in equation-1 and step-1 of section-3.2.2. A weighted graph has then converted to Laplace graph(L) as in step-2 of section-3.2.2. Secondly, a spatial-potential matrix (V) has constructed as in step-3. In step-5, the spatial-matrix(V) has fused with Laplace matrix (L) to utilize the spatial potential of spatial-spectral HD image datacube. For example, the Laplace graph and spatial potential graph of combined spatial-spectral datacube obtained from AlexNet architecture have shown in figure-3.6(a), (b), (c), and figure-3.6(d), (e), (f). Figure-3.7(a), (b), (c) shows the data distribution in the first three dimensions of the features in the graph-based embedded manifold for AlexNet CNN. It has noticed that graph-based manifold embedding leads to high separability of data points in the new manifold. Data in a new manifold is highly separable as well as similar to

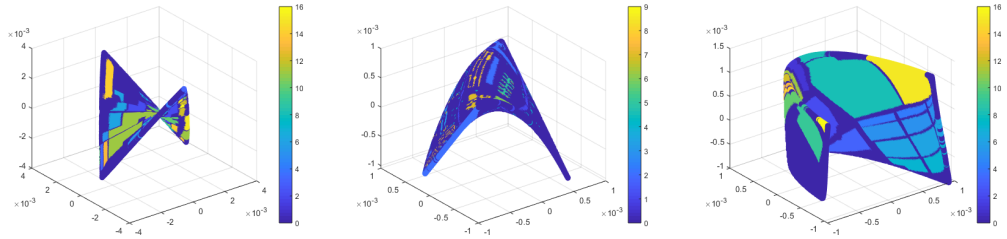
the original ground truth. It can be observed that the data points are very less separable when proposed manifold embedding has not applied, as shown in figure-3.5. Similarly, other manifold embedding results for VGG-16 and VGG-19 architecture has shown in figure-3.7(d), (e), (f), and figure-3.7(g), (h), (i) respectively. Subsequently, we have applied LDA on embedded dataset to obtain global posterior probability denoted as

$$p_g(f_i|c_i) \; \forall \; i \; \varepsilon \; k$$

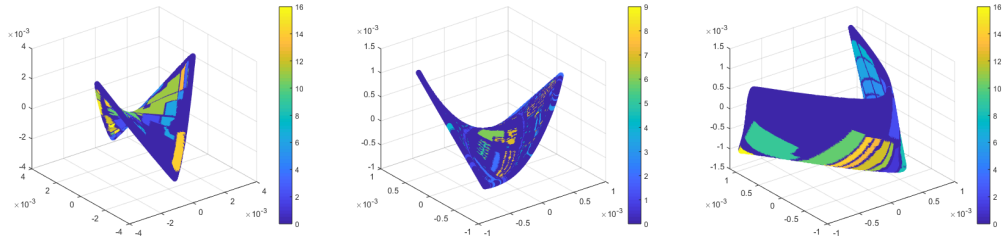
in step-7 of section-3.2.2. The results for global probability estimation has shown in figure-3.8(c) and 3.8(f) for IP and PU datasets.



(A) IP-AlexNet (B) PU-AlexNet (C) SV-AlexNet



(D) IP-VGG-16 (E) PU-VGG-16 (F) SV-VGG-16



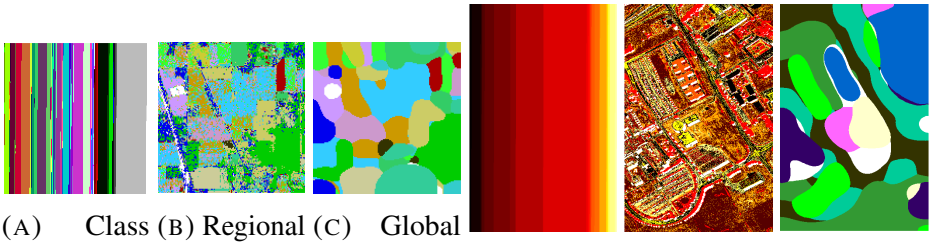
(G) IP-VGG-19 (H) PU-VGG-19 (I) SV-VGG-19

FIGURE 3.7: Data distribution on first three Embedded Manifold for spatial-spectral combined datacube obtained from graph based manifold embedding of IP, PU, SV datasets

3.3.4 Regional Probability estimate with multinomial regression

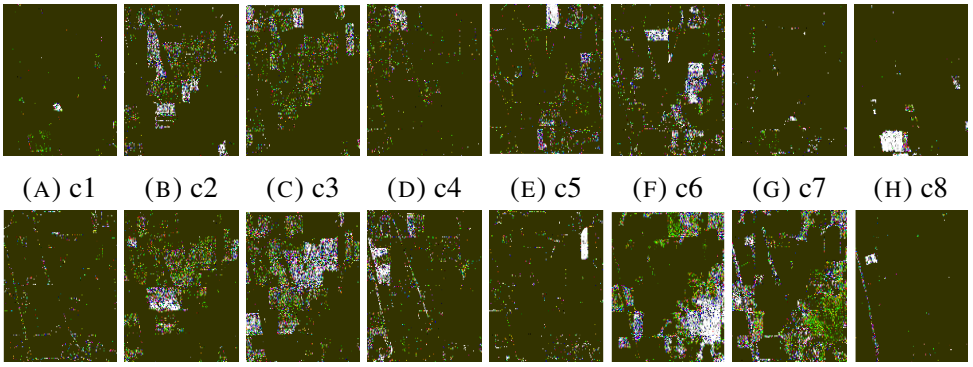
In this section, the regional probability of pixels has calculated. The regional probability has obtained in two steps. **In the first step**, we have computed the posterior probability of pixels by using Hysime based subspaceMLR method. It has found that the subspaceMLR method is very efficient in dealing with mixed class pixels. **In the second step**, a class mixing image has been generated for mixing parameter I=2, .i.e. two mixed-classes per pixel. The class probability has been set to zero for local probability estimation. Further, the subspaceMLR method has applied to the mixed pixels of the obtained class map. The posterior probability of this step has denoted as the regional probability of pixels, as discussed earlier. It is obvious that if mixing parameter 'I' and the number of classes are the same, then there is no mixing in pixel.

Figure-3.8 (a) and 3.8(d) denotes the class mixing image for IP and PU datasets.



(A) Class mixing (B) Regional probability (C) Global probability (D) Class mixing (E) Regional probability (F) Global probability

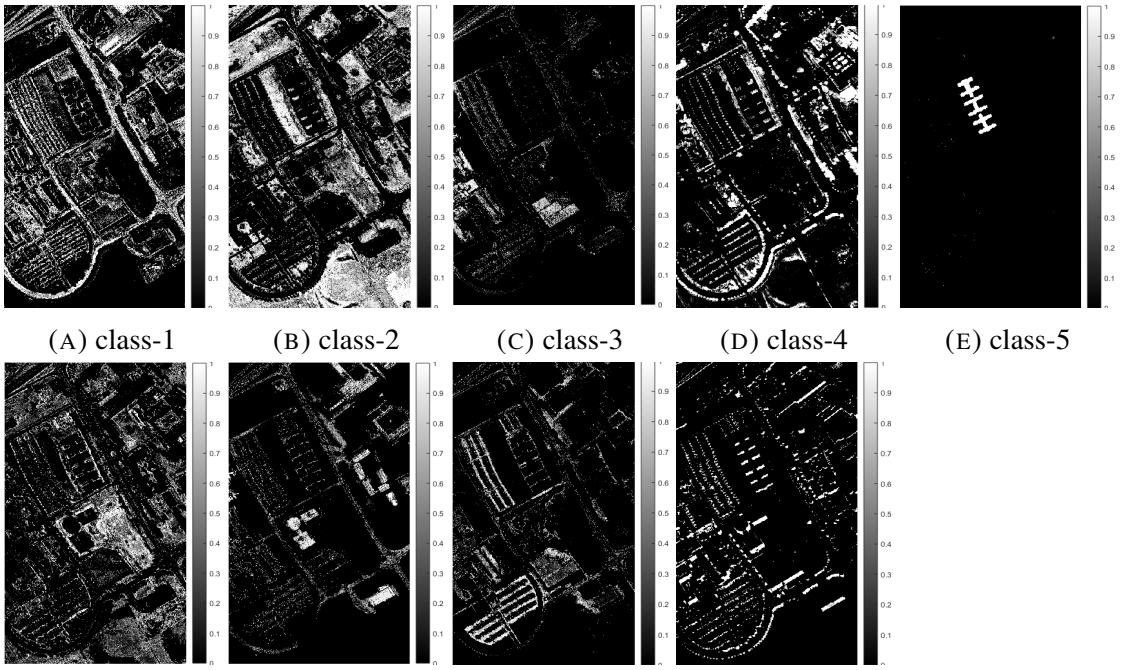
FIGURE 3.8: (a,d)-class mixing images for I=2, (b,e)- classification results obtained from Regional probability with (OA=72.95 %) and (OA=73.03 %), (c,f)- Gobal probability for spatial-spectral datacube of IP and PU respectively



(A) c1 (B) c2 (C) c3 (D) c4 (E) c5 (F) c6 (G) c7 (H) c8

(I) c09 (J) c10 (K) c11 (L) c12 (M) c13 (N) c14 (O) c15 (P) c16

FIGURE 3.9: Class-wise Regional probability images for spectral datacube of IP



(A) class-1 (B) class-2 (C) class-3 (D) class-4 (E) class-5

(F) class-6 (G) class-7 (H) class-8 (I) class-9

FIGURE 3.10: class mixing image(a) and Regional probability images(b to j) for spectral datacube of PU

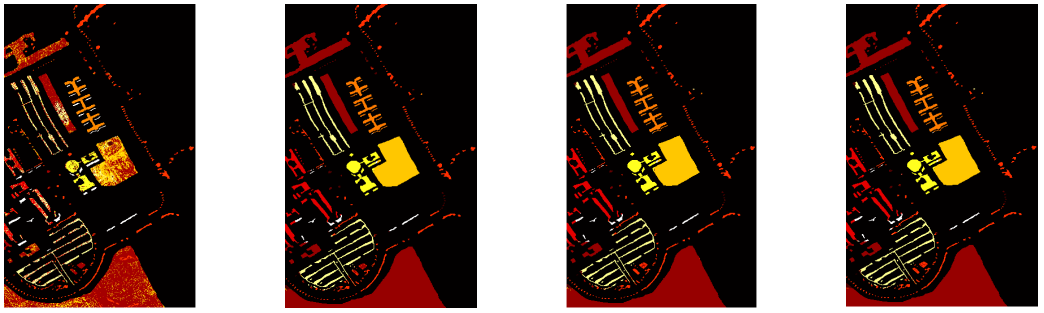
Figure-3.8(b) and 3.8(e) have depicted the classification results obtained from regional probability estimates for IP and PU datasets. Figure-3.9(a) to 3.9(p) and figure-3.10(a) to 3.10(i) have denoted the regional probability estimates for IP and PU datasets, respectively. Regional probability for Salinas Valley(SV) datasets has obtained with the similar procedure discussed above.

3.3.5 Fusion of Global and Regional information

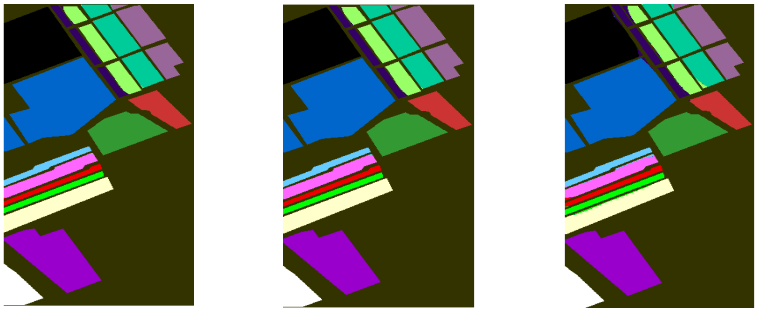
In this section, we have fused the Regional and Global probabilities with the help of equation-3.6. We have used 10366, 12837, 54129 samples of IP(Indian Pines), PU(Pavia University), and SV(Salinas Valley) datasets respectively in 16, 9, 16 classes to obtain regional and global probabilities. These datasets are commonly considered the benchmark images. It has found that these HD-HSIs(High dimensional hyperspectral images) contain the noisy and mixed class pixels in the image, which is a favorable scenario for regional probability computation since it relies upon class mixing in pixels. Therefore, our proposed method is gaining a significant increase in accuracy on these datasets. The subspaceMLR

method has applied for the calculation of regional probability, for only 10 %, 4%, and 6% samples of IP, PU, SV dataset, respectively, in training. The complete dataset has used for regional probability calculation. Overall accuracy(OA) achieved by the regional probability of IP and PU datasets is 73.12% and 67.15%, as shown in Table 3.2 and 3.3. Figure-3.11(a) and 3.11(e) has depicted the final classification results obtained from the regional probability estimate. Global posterior probability has computed for complete datasets using LDA learners. The global and regional probability has been integrated using regularizers λ_1 and λ_2. Value of regularizer λ_2 has been chosen 0.6, 0.5 and 0.5 for IP, PU, SV dataset respectively. When λ_1=0, then the regional probability is effective to obtain final probability, and λ_2=0 leads to the global estimate only. The result obtained from regional and global probability fusion for IP, PU, and SV datasets has shown in Table 3.2, 3.3, and 3.4. The regional-global fusion of AlexNet features for IP, PU, and SV datasets has shown in Figure-3.11(b), (e), (i). The fusion of VGG-16 features for IP, PU, and SV datasets has depicted in Figure-3.11(c), (f), (j). For VGG-19, the fusion results have shown in figure-3.11(d), (h), and (k), respectively. In table-3.2, the regional-global probability fusion of VGG-16 features with IP dataset is achieving maximum accuracy, .i.e. 97.38%, than other cases. The regional-global fusion of AlexNet and VGG-19 features for IP is yielding 96.52% and 95.36% accuracy. Our fusion-based method has compared with some previously proposed efficient methods. Our proposed method is achieving 10%, 9%, 11%, 3% higher accuracy than LBP-ELM [58], SVM-RFS [65], 1-D-CNN [53], and CNN-PPF [59] methods, which is a significant improvement for IP dataset. The classification results for the regional-global fusion of PU dataset has shown in Table 3.3. It has found that regional-global fusion for VGG-19 features has achieved the maximum accuracy, i.e., 95.10%, for PU dataset among the proposed methods. However, previously proposed method CNN-PPF is achieving maximum accuracy .i.e., 96.48, in this case. Regional-global methods for AlexNet and VGG-16 is exhibiting 95.09% and 95.01% accuracy, which is higher than most of the previously proposed methods. Regional-global fusion for VGG-19 features is achieving 6%, 4%, 3% higher accuracy than LBP-ELM, SVM-RFS, 1-D-CNN methods. The regional-global classification results obtained for the Salinas valley dataset(SV) dataset has shown in Table 3.4. The regional-global fusion for VGG-16 and SV features has carried out the maximum overall accuracy, .i.e. 99.88%. Other results for AlexNet and VGG-19 is exhibiting the 99.85% and 99.68% overall accuracy. Our regional-global VGG-16 methods have achieved 7%, 6%, 10%, 3% higher accuracy than previously proposed LBP-ELM, SVM-RFS, 1-D-CNN methods, respectively.

(A) Regional probability IP (OA= 72.95%) (B) Regional-Global probability IP-AlexNet (OA= 96.92%) (C) Regional-Global probability IP-VGG-16 (OA= 97.38%) (D) Regional-Global probability IP-VGG-19 (OA= 95.36%)



(E) Classification using Regional probability for PU using spectral (OA= 67.15%) (F) Regional-Global probability for PU-AlexNet (OA= 95.09%) (G) Regional-Global probability for PU-VGG-16 (OA= 95.01%) (H) Regional-Global probability for PU-VGG-19 (OA= 95.10%)



(I) Classification using Regional-Global probability SV-(OA= 99.85%) (J) Regional-Global probability SV-VGG-16 (OA= 99.88%) (K) Regional-Global probability SV-VGG-19 (OA= 98.68%)

FIGURE 3.11: Regional and global probability fusion for IP,PU,SV datasets for AlexNet, VGG-16 and VGG-19

TABLE 3.2: Result of spatial-spectral data on Indian Pines Image

Classes	LBP-ELM[58]	SVM-RFS [65]	1-D-CNN [53]	CNN-PPF [59]	Regional probability	Alexnet(Regional-Global fusion)	vgg16(Regional-Global)	vgg19(Regional-Global)
alfalfa	86.06	88.73	78.58	92.99	66.67	96.30	96.15	96.30
corn_no till	88.19	91.20	85.23	96.66	56.77	97.85	98.29	95.61
corn_min till	96.07	97.52	95.75	98.58	59.97	94.73	95.28	90.62
corn	99.73	99.86	99.81	100	71.93	99.42	100	99.42
grass/pasture	100	100	99.64	100	87.59	95.08	95.31	93.46
grass/tree	90.02	91.67	89.63	96.24	95.70	94.81	95.02	95.22
grass/pasture-mowed	71.00	78.79	81.55	87.80	84.62	100	100	100
Hay-windrowed	95.62	93.76	95.42	98.98	97.16	100	100	100
oats	98.66	98.74	98.59	99.81	100	100	100	100
soyabean_no till	-	-	-	-	55.89	91.47	92.45	89.74
soyabean_min till	-	-	-	-	70.44	98.81	99.32	97.54
soyabean_clean till	-	-	-	-	75.28	90.71	92.07	83.43
wheat	-	-	-	-	100	100	100	100
woods	-	-	-	-	91.35	99.92	99.92	99.92
bldg-grass tree	-	-	-	-	56.91	100	100	99.68
stone-steel tower	-	-	-	-	82.86	100	100	100
OA	**87.33**	**89.83**	**86.44**	**94.34**	**73.12**	**96.92**	**97.38**	**95.36**
AA	-	-	-	-	**78.32**	**97.40**	**97.70**	**96.31**

TABLE 3.3: Result of spatial-spectral data on Pavia University Dataset

Classes	LBP-ELM [58]	SVM-RFS [65]	1-DCNN [53]	CNN-PPF [59]	Regional probability result	Alexnet(Regional-global)	vgg16(Regional-global)	vgg19(Regional-global)
alfalfa	81.32	87.95	88.38	97.42	38.18	91.26	90.91	92.80
corn_no till	90.91	91.17	91.27	95.76	73.31	99.04	98.53	98.37
corn_min till	85.09	86.99	85.88	94.05	54.92	96.03	96.36	96.17
corn	96.61	95.50	97.24	97.52	75.81	82.29	84.86	83.60
grass/pasture	99.63	99.85	99.91	100	100	100	100	100
grass/tree	94.33	94.31	96.41	99.13	59.74	99.38	99.24	99.38
grass/pasture-mowed	95.94	94.74	93.62	96.19	91.62	98.84	99.71	99.43
Hay-windrowed	82.65	85.89	87.45	93.62	71.48	96.67	99.67	96.19
oats	99.79	99.89	99.57	99.60	100	31.91	32.05	31.91
OA	**89.86**	**91.10**	**92.27**	**96.48**	**67.15**	**95.09**	**95.01**	**95.10**
AA	-	-	-	-	**73.90**	**88.70**	**99.68**	**88.65**

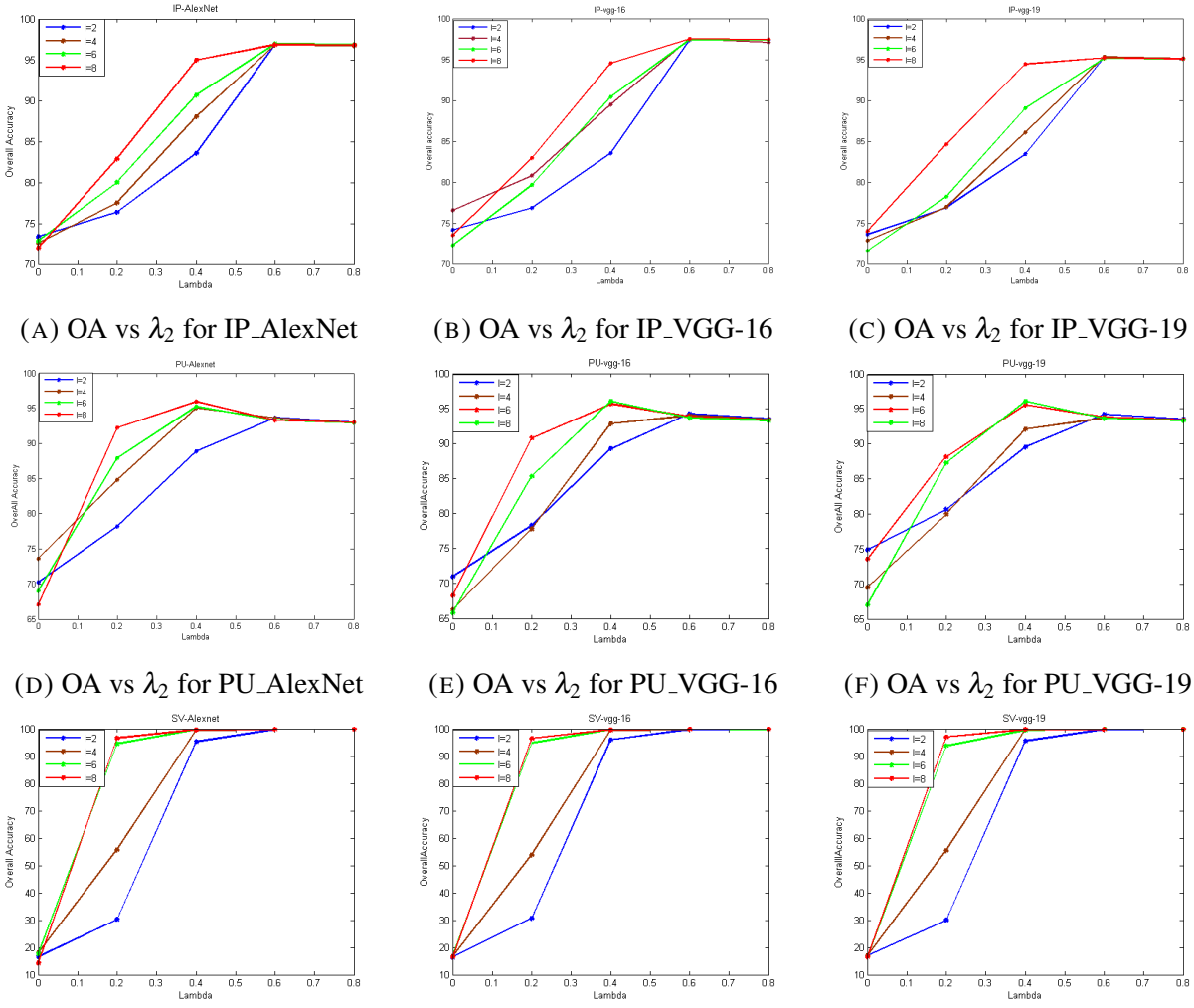
TABLE 3.4: Result of spatial-spectral data on Salinas valley Image

Classes	LBP-ELM [58]	SVM-RFS [65]	1-D CNN [53]	CNN-PPF [59]	Alexnet(Regional-global)	vgg16(Regional-global)	vgg19(Regional-global)
alfalfa	99.75	99.55	97.34	100	99.79	100	99.72
corn_no till	99.87	99.92	99.29	99.88	100	100	100
corn_min till	99.60	99.44	96.51	99.60	100	100	99.77
corn	99.64	99.86	99.66	99.49	100	100	100
grass/pasture	98.81	98.02	96.97	98.34	97.91	98.28	93.30
grass/tree	99.67	99.70	99.60	99.97	99.78	99.92	97.91
grass/pasture-mowed	99.66	99.69	99.49	100	99.79	99.85	99.49
Hay-windrowed	84.04	84.85	72.25	88.68	100	100	99.58
oats	99.89	99.58	97.53	98.33	100	100	100
soyabean_no till	95.03	96.49	91.29	98.60	100	100	95.21
soyabean_min till	96.82	98.78	97.58	99.54	99.88	99.88	99.53
soyabean_clean till	100	100	100	100	99.41	99.41	93.15
wheat	98.25	99.13	99.02	99.44	100	100	99.72
woods	97.94	98.97	95.05	98.96	100	100	100
bldg-grass tree	72.96	76.38	76.83	83.53	100	100	98.82
stone-steel tower	99.06	99.56	98.94	99.31	100	100	100
OA	**92.42**	**93.15**	**89.28**	**94.80**	**99.85**	**99.88**	**98.68**
AA	-	-	-	-	**99.80**	**99.83**	**98.51**

3.3.6 Effect of λ_2 on OA

In this section, an assessment of the regularizer effect λ_2 on overall accuracy (OA) has described. The λ_1 and λ_2 regulates the Regional and Global knowledge used in the proposed method. Parameter 'I' controls the number of classes mixed for regional probability computation. For example, I=4 corresponds to the mixing of four classes in each pixel of the image. The value of λ_2 has increased by a step size of 0.1 on which overall accuracy has measured. The OA vs. λ_2 behavior has obtained for various values of $I\varepsilon(2,8)$, as shown in Figure-3.12. Figure-3.12(a), (b), (c) shows the classification results of the IP dataset received by the fusion of regional-global probabilities. Figure-3.12(d), (e), (f) shows the OA vs λ_2 plot of PU dataset and (g), (h), (i) of SV dataset. It has noted that the OA is increasing with λ_2 and achieves the peak at $\lambda_2 = 0.6$ for the IP dataset. Subsequently, the graph is decreasing to λ_2=0.6. Figure-3.12(d), (e), (f) are showing the

graph for the PU dataset that shows the maximum OA at λ_2=0.4. The maximum OA is achieved at λ_2=0.6 for SV dataset, as shown in figure-3.12 (g), (h), (i).



(A) OA vs λ_2 for IP_AlexNet (B) OA vs λ_2 for IP_VGG-16 (C) OA vs λ_2 for IP_VGG-19

(D) OA vs λ_2 for PU_AlexNet (E) OA vs λ_2 for PU_VGG-16 (F) OA vs λ_2 for PU_VGG-19

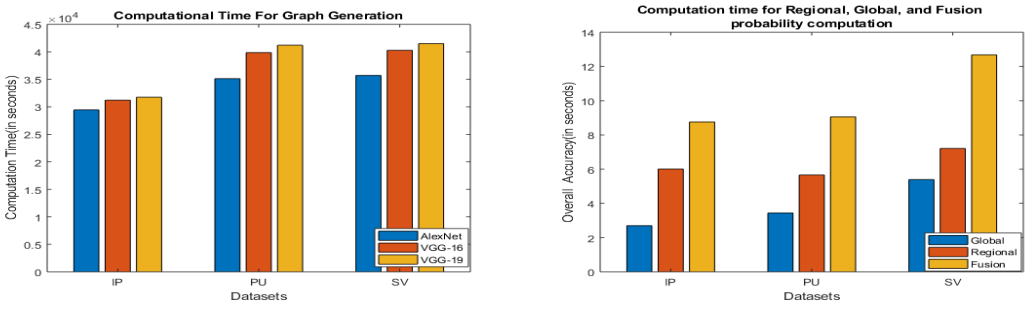
(G) OA vs λ_2 for SV_AlexNet (H) OA vs λ_2 for SV_VGG-16 (I) OA vs λ_2 for SV_VGG-19

FIGURE 3.12: OA vs λ_2 for Regional-global probability fusion of spatial-spectral IP, PU, SV datasets

3.3.7 Computational Complexity

In this section, we have evaluated the Computational complexity and execution time of our proposed method. Firstly, we have calculated the computation time for the graph-based manifold learning method. Figure-3.13(a) has shown the computational time for the generation of a dense 9-nearest neighbor graph by using the features of AlexNet, VGG-16,

and VGG-19 architectures. It can be observed that features extraction and probability computation time for VGG-19 architecture is maximum for IP, PU, and SV datasets. Our proposed method has divided into three main steps: (1) Global probability (2) Regional probability (3) Fusion of probabilities, as discussed in section-3. Global probability has computed by LDA having time complexity$O(N \times d^2)$ ε $d : N > d$, where N is the number of samples and d is the dimension of selected features. In the worst case, the time complexity of global probability computation is $O(d^3)$. The regional probability computation exhibits two steps:(1) class mixing image generation and (2) regional probability computation using linear regression. The complexity of class mixing image generation is $O(k \times N)$, where k is the number of mixing samples, and N is the total number of samples. Secondly, the regional probability has computed on all the mixed samples. Therefore, the probability of regional probability computation is $O(\frac{N^2}{k})$, where k is the mixing samples, and N is the total number of samples. Finally, the overall worst-case complexity of fusion is $O(N^2) + O(d^3)$. Figure-3.13(b) has shown the computation time of regional, global, and fusion probabilities computation. We have observed that the overall probability fusion process takes maximum time in VGG-19 features for IP, PU, and SV datasets.



(A) Computational time for k-NN graph generation

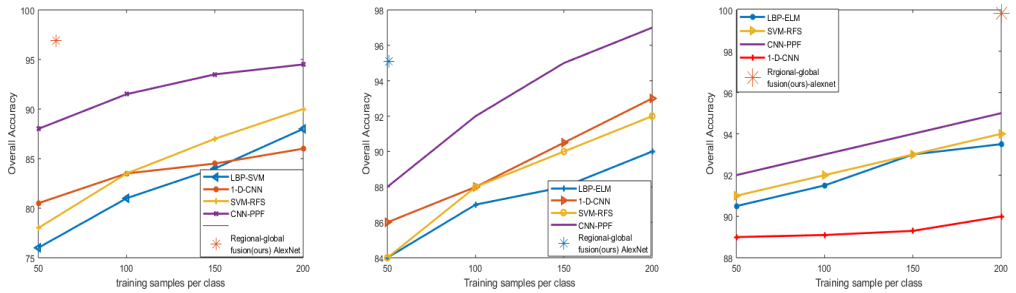
(B) Computation Time for Regional Global and Fusion probability computation

FIGURE 3.13: Computation Time for IP, PU, SV datasets

3.3.8 Comparison with other methods

In this section, the proposed method is compared with recently proposed state-of-art algorithms. In the past state-of-art methods, the overall accuracy has plotted against the training samples per class with a step size of 50 in an interval of 0 to 200. Our deep

spatial-spectral fusion method has trained on 60, 50, and 200 samples per class for regional probability calculation of IP, PU, and SV datasets. In contrast, the global probability has calculated on the complete dataset. The overall accuracy of the proposed method has denoted as an asterisk along with the OA vs. training sample graph of previously proposed methods, i.e., LBP-SVM, 1-D CNN, SVM-RFS, CNN-PPF. The comparative graphs have shown in figure-3.14(a), (b), (c) for IP, PU, and SV datasets, respectively, for AlexNet architecture. It has noticed that our regional and global probabilistic fusion algorithm is outperforming the previously proposed methods with a significant margin for 60, 50, and 200 training samples per class.



(A) (B) (C)

FIGURE 3.14: Overall Accuracies for different models in (a) Indian Pines(b)Pavia University(c) Salinas Valley Dataset

3.4 Conclusion

In this work, we have introduced a two-stage probabilistic fusion method. In the first stage, Deep learning-based spatial features information is extracted and fused with spectral knowledge using a graph-based manifold learning method. Deep spatial features are concatenated with the spectral feature to obtain more informative data-cube for manifold learning. In stage-2, the regional probability has computed with a class mixing image derived from the original HD image by considering the appropriate class mixture at each pixel. The regional probability calculation reduces the local noise and effect of insignificant class mixtures in each pixel. The number of mix-classes per pixel has controlled by a parameter 'I' in class mixing image. This stage incorporates the well-known effect of spectral mixing in the dataset. Subsequently, the Regional and global probability has fused

with regularizers λ_1 and λ_2. The experimental results have shown that the proposed method has outperformed the previously proposed methods. Deep spatial features and regional probability estimates are increasing the classification accuracy significantly. Therefore in the future, we will incorporate the deep features with multi-source features using graph-based manifold learning operations.