Remote Sensing Letters

ISSN: 2150-704X (Print) 2150-7058 (Online) Journal homepage: http://www.tandfonline.com/loi/trsl20

High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder

Weixun Zhou, Zhenfeng Shao, Chunyuan Diao & Qimin Cheng

To cite this article: Weixun Zhou, Zhenfeng Shao, Chunyuan Diao & Qimin Cheng (2015) High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder, Remote Sensing Letters, 6:10, 775-783, DOI: 10.1080/2150704X.2015.1074756

To link to this article: https://doi.org/10.1080/2150704X.2015.1074756

Remote Sensing Letters, 2015
Vol. 6, No. 10, 775–783, http://dx.doi.org/10.1080/2150704X.2015.1074756

Taylor & Francis
Taylor & Francis Group

High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder

Weixun Zhou[a,b], Zhenfeng Shao[a,b]*, Chunyuan Diao[c], and Qimin Cheng[d]

[a]*State key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China;* [b]*Collaborative Innovation Centre for Geospatial Technology, Wuhan, China;* [c]*Department of Geography, University at Buffalo, The State University of New York, Buffalo, NY, USA;* [d]*School of Electronic Information and Communications, Huazhong University of Science & Technology, Wuhan, China*

(*Received 7 April 2015; accepted 14 July 2015*)

An unsupervised feature learning framework based on auto-encoder is proposed to learn sparse feature representations for remote-sensing imagery retrieval in this letter. The low-level feature descriptors are extracted and exploited to learn a set of feature extractors, which are then used to encode the low-level feature descriptors to generate new sparse features. The learned feature representations are applied to aerial images randomly selected from the University of California Merced data set. The results indicate that the performance of our proposed framework is comparable or superior to that of the state-of-the-art method. The framework is proved to be an effective approach to manage the huge volume of remote-sensing data and to retrieve the desired remote-sensing imagery.

1. Introduction

With the development of high-resolution satellite sensors, a huge volume of high-resolution remote-sensing imagery becomes available. As a result, content-based remote sensing imagery retrieval (CBRSIR) technology has drawn more public attention in recent years. Most of the works in the image retrieval literature focus on feature extraction because retrieval performance greatly depends on the power of feature representations.

In CBRSIR methods, low-level features can be primarily categorized into global and local descriptors. With respect to global descriptors, spectral information, shape features and texture features are commonly used. Specially, texture features, such as Gabor features, have been widely investigated due to its periodicity, coarseness and directionality characteristics (Aptoula 2014). Newsam et al. (2004) explored Gabor texture features to analyse and manage large collections of satellite imagery. Unlike global descriptors, local descriptors focus on salient regions or points. Yang and Newsam (2013) exploited Scale Invariant Feature Transform (SIFT) to generate visual words for bag of visual words (BOVW) representation and explored BOVW for aerial image retrieval. It concludes that BOVW representation outperforms simple statistics, homogeneous texture and colour histogram for most image classes due to its local, invariance and robust properties. Chen et al. (2011) compared a variety of global and local descriptors for very-high-resolution image scene classification, and demonstrated

*Corresponding author. Email: shaozhenfeng@whu.edu.cn

that SIFT descriptor achieved the best performance among all the evaluated features. However, all of the features, including global and local descriptors, are hand-crafted features where it is usually laborious and time-consuming to design these informative and powerful feature representations.

Considering the drawbacks of global and local descriptors, there exist great demands to concentrate on features that are learned in an unsupervised way. Hinton and Salakhutdinov (2006) introduced a layer-wise learning algorithm to initialize the weights of deep auto-encoder networks, which makes the training of deep neural networks much easier. Since then, unsupervised feature learning has been widely used for image classification and object recognition. In some recent works, unsupervised feature learning was even successfully applied to remote-sensing scene classification and object detection. In particular, Cheriyadat (2014) proposed an unsupervised feature learning method through combining dense SIFT and Orthogonal Matching Pursuit (Pati, Rezaiifar, and Krishnaprasad 1993) for aerial scene classification. The proposed method essentially belongs to sparse coding and outperforms state-of-the-art methods BOVW, spatial pyramid matching kernel (SPMK) (Lazebnik, Schmid, and Ponce 2006) as well as the spatial extension of BOVW and SPMK (SPCK++) (Yang and Newsam 2011). In a recent work by Zhang, Du, and Zhang (2015), the sparse auto-encoder network, an unsupervised feature learning method, was trained with both salient and non-salient patches to learn feature representations for aerial scene classification and achieved better performance than the approach introduced in Cheriyadat (2014). However, the saliency detection algorithm utilized to extract salient patches suffers from complexity and inefficiency. Many parameters such as image patch size, sparsity penalty, the stride for feature convolution and the window size for feature pooling should be investigated. Some other similar works can be found in Cheng et al. (2015) and Han et al. (2014, 2015).

Although some recent studies have focused on unsupervised feature learning methods for remote-sensing tasks, few works have been done on CBRSIR. The motivation of this study lies in developing an unsupervised feature learning framework (UFLF) based on auto-encoder to learn sparse feature representations for CBRSIR. The proposed framework requires less parameter than sparse auto-encoder and avoids feature convolution by using low-level feature descriptors instead of image patches for training. For feature pooling, we use average pooling that has no window size parameter. In terms of activation function of the hidden layer, rectified linear (ReL) function rather than conventional sigmoidal function is selected, since it can enforce sparsity on hidden units and reduce gradient vanishing problem (Glorot, Bordes, and Bengio 2011).

2. Methodology

The proposed UFLF consists of four steps: (1) feature extraction, (2) unsupervised feature learning, (3) feature encoding, and (4) sparse feature generation and pooling. The initial step of UFLF is to extract low-level feature descriptors from the training images. During feature learning, the extracted feature descriptors are fed into auto-encoder network to generate a set of feature extractors. Once the auto-encoder network is trained, the learned features can be computed by feature encoding. In the final step, we apply soft threshold function to generate sparse features, which are then pooled to generate the final feature representation. Figure 1 shows the overall architecture of UFLF.

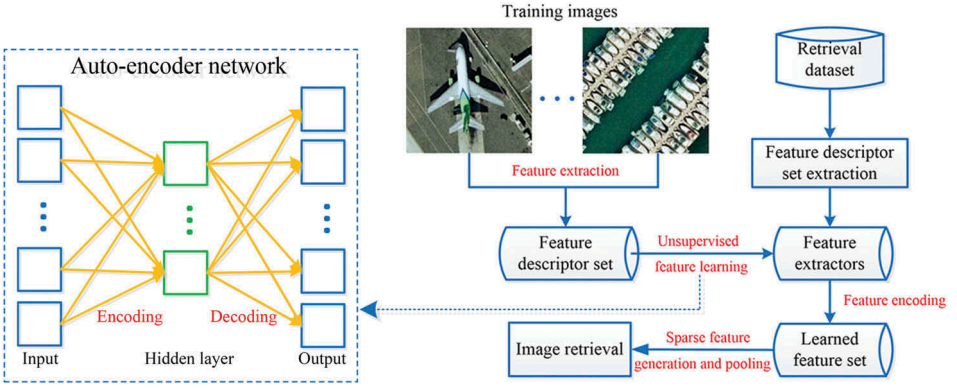

Figure 1. Overview of the proposed remote-sensing imagery retrieval framework.

2.1. *Feature extraction*

In most cases, representative image patches sampled from the entire image are used to train the auto-encoder network to compute the feature extractors. These learned feature extractors can be viewed as a feature mapping function that maps an image patch to a new feature representation. To obtain the feature representation of the entire image, feature convolution is usually needed. However, feature convolution has low computational efficiency and the stride for feature convolution should be defined.

In this study, SIFT and dense SIFT descriptors extracted from the training images are used to train the auto-encoder network. The motivation derives from two aspects. On the one hand, a SIFT or dense SIFT descriptor corresponds to a local region but performs better than raw pixels (the image patch). On the other hand, since the image is represented by a set of local descriptors, we can obtain the new feature representations of the entire image without feature convolution.

Once the feature descriptors of an image are extracted, it is then represented by a feature matrix $\mathbf{X}_t=[\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_n]$, where $\boldsymbol{x}_i(\ i=1,2...,n)$ is one 128-dimensional feature vector, t is the image index and n is the number of feature descriptors.

2.2. *Unsupervised feature learning*

The aim of the auto-encoder network is to learn a compressed feature representation from high dimensional feature space by minimizing the reconstruction error between the input and output layers. The number of nodes in the input layer is equal to that of the output layer. To reduce the dimensionality of data, the auto-encoder network reconstructs the feature descriptor set with fewer nodes in the hidden layers. The activations of the hidden layer are usually regarded as the compressed features. In this letter, we use a three-layer auto-encoder network consisting of two stages, encoding and decoding. The dotted rectangle in Figure 1 shows the structure of the auto-encoder network.

Describe $\mathbf{X}=[\mathbf{X}_1,\mathbf{X}_2,...,\mathbf{X}_n]=[\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_N]$ as being the matrix representing the stacked feature descriptors extracted from the training images, where n is the number of training images and N is the number of feature descriptors. The feature matrix $\mathbf{X}$ is normalized by subtracting the mean, and whitened by Zero Component Analysis (ZCA) transform. Normalization and whitening are computed using Equations (1) and (2), respectively.

$$\bar{\mathbf{X}} = \mathbf{X} - \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \tag{1}$$

$$\mathbf{X}_{\text{whitening}} = \frac{\mathbf{U}\mathbf{U}^{\mathrm{T}}}{\sqrt{\mathbf{S} + \varepsilon}} \tag{2}$$

Here, $\bar{\mathbf{X}}$ is the result of normalization and $\mathbf{X}_{\text{whitening}}$ is the result of whitening. $\mathbf{U}$ and $\mathbf{S}$ (a diagonal matrix) are matrices consisting of the eigenvectors and the eigenvalues of the covariance matrix of $\bar{\mathbf{X}}$, respectively. ε is a constant close to 0, and T is transpose operation.

During the encoding stage, the input $\mathbf{X}_{\text{whitening}}$ is mapped to the activation value $\mathbf{h}_1$ of the hidden layer through the ReL activation function.

$$\mathbf{h}_1 = f(\mathbf{W}_1\mathbf{X}_{\text{whitening}} + \boldsymbol{b}_1) \tag{3}$$

Here, $\mathbf{W}_1$ and $\boldsymbol{b}_1$ are the weight matrix and the bias term of the encoding stage, respectively. $f(x)$ is the ReL activation function of the hidden units and it is defined as $f(x) = \max(0, x)$. During the decoding stage, the output $\mathbf{h}_2$ of the network is obtained by mapping $\mathbf{h}_1$ through a non-linear activation function.

$$\mathbf{h}_2 = g(\mathbf{W}_2\mathbf{h}_1 + \boldsymbol{b}_2) \tag{4}$$

Here, $\mathbf{W}_2$ and $\boldsymbol{b}_2$ are the weight matrix and the bias term of the decoding stage, respectively. $g(x)$ is the softplus activation function of the output layer and it is defined as $g(x) = \ln(1 + \mathrm{e}^x)$. The sigmoidal function $f(x) = 1/(1 + \mathrm{e}^{-x})$ and the linear function $f(x) = x$ are also two commonly used activation functions.

The features are learned by minimizing the overall reconstruction error in Equation (5).

$$L(\mathbf{W}, \mathbf{b}) = \frac{1}{2}\left\|\mathbf{X}_{\text{whitening}} - \mathbf{h}_2\right\|^2 + \frac{\lambda}{2}\|\mathbf{W}\|^2 \tag{5}$$

Here, the first term is a square error term, and the second term is a regularization term that helps prevent over-fitting. λ is the weight decay parameter. The network is trained by optimizing Equation (5) with respect to $\mathbf{W}_1, \mathbf{W}_2, \boldsymbol{b}_1, \boldsymbol{b}_2$. In this letter we use Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) to optimize this problem.

2.3. *Feature encoding*

Given the weight $\mathbf{W}_1$ and the bias $\boldsymbol{b}_1$, we encode the low-level feature descriptors to generate new feature representations using Equation (6).

$$\mathbf{Y} = f(\mathbf{W}_1\mathbf{X}_t + \boldsymbol{b}_1) \tag{6}$$

Here, $\mathbf{Y}$ are the learned feature representations. $f(x)$ is the ReL activation function, and $\mathbf{X}_t$ is the feature matrix of the image with index t, which is preprocessed with the same mean and whitening matrices as those used in the auto-encoder training process.

2.4. Sparse feature generation and pooling

Sparsity can be defined as having few non-zero components. In this letter, we obtain sparse feature representations in two steps. We first use ReL function as the activation function to impose sparsity on the hidden units. Then we proceed to enforce more sparsity using the soft threshold function. Given the learned feature set $\mathbf{Y}$, we generate a sparse feature representation $\mathbf{Z}$ with Equation (7).

$$\mathbf{Z} = [\mathbf{Z}^+, \mathbf{Z}^-] \tag{7}$$

Here, $\mathbf{Z}^+ = \max\{0, \mathbf{Y} - \alpha\}$ and $\mathbf{Z}^- = \max\{0, \alpha - \mathbf{Y}\}$ are the positive and negative weights above and below the threshold α that enforces sparsity. '$\mathbf{Y} - \alpha$' or '$\alpha - \mathbf{Y}$' means the subtraction operation between α and each element of $\mathbf{Y}$, and the result is also a matrix.

With the sparse feature set $\mathbf{Z}$, we then apply average pooling to obtain the final feature representation for image retrieval. The equation is shown in Equation (8).

$$\mathbf{F} = [\boldsymbol{f}^+, \boldsymbol{f}^-] = \frac{1}{N}\sum_{i=1}^{N}[\boldsymbol{z}_i^+, \boldsymbol{z}_i^-] \tag{8}$$

Here, $\mathbf{F}$ is the final feature representation. $\boldsymbol{z}_i^+$ and $\boldsymbol{z}_i^-$ are the *ith* column vectors of $\mathbf{Z}^+$ and $\mathbf{Z}^-$, $\boldsymbol{f}^+$ and $\boldsymbol{f}^-$ are the pooled feature vectors of $\mathbf{Z}^+$ and $\mathbf{Z}^-$, respectively.

3. Experiments

In this section the University of California (UC) Merced data set is used in our experiments, and the performance of our UFLF is demonstrated.

3.1. Data set and experimental setup

3.1.1. Data set

UC Merced data set contains 21 challenging scene categories with 100 samples per class. Each image has 256 × 256 pixels with a resolution of 30 cm. For computational efficiency, we randomly selected 10 image categories to constitute the retrieval data set. Figure 2 shows some example images of the selected image categories.

3.1.2. Experimental setup

In our experiments the training images are randomly selected from each image category with 50 images per category for SIFT descriptors, and 25 images per category for dense SIFT descriptors. For SIFT we use the original algorithm by Lowe (2004), and for dense SIFT we set the sampling window size to 16 × 16 and the step size to 8 pixels. For ZCA whitening we set the constant ε to 0.1, and for the weight decay parameter we set λ to 0.001. For the number of hidden units we set it to 400 and for the sparsity parameter α several values (0.035, 0.05, 0.2, 0.4, 0.6, 0.8 and 1.0) are considered.

Here we also explore different configurations of BOVW representation against which our UFLF is compared. For BOVW representation, *k*-means clustering using Euclidean distance measure is applied to generate the codebook with different number of clusters (200, 400, 600, 800, 1000, 1200 and 2000).

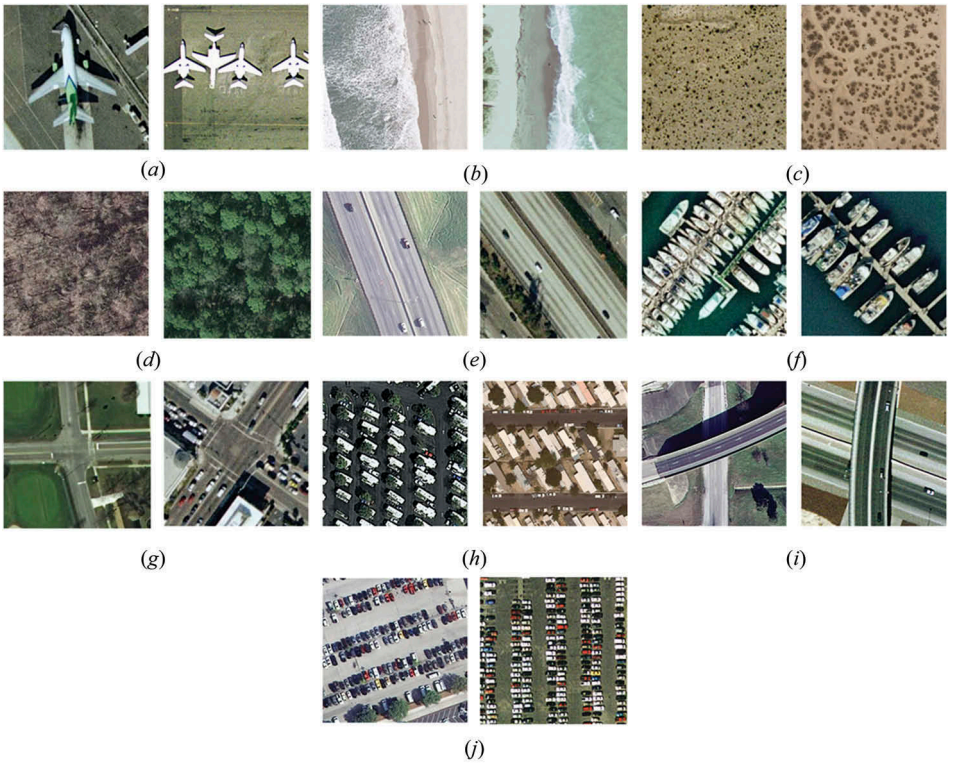

Figure 2. Some example images from UC Merced data set (Yang and Newsam 2013): (*a*) airplane, (*b*) beach, (*c*) chaparral, (*d*) forest, (*e*) freeway, (*f*) harbour, (*g*) intersection, (*h*) mobile home park, (*i*) overpass and (*j*) parking lot.

To measure the similarity between the query image and other images in the database, L1 (City Block distance) and L2 (Euclidean distance) distances are used to evaluate the similarity for sparse features generated by our UFLF and histogram intersection is used for BOVW representation.

3.2. Results

To evaluate the performance of UFLF, we conduct several experiments. Figure 3(*a*) shows the average precision over all 10 classes for the two feature extraction strategies with varying sparsity values. UFLF based on SIFT descriptors is shown to perform better than UFLF based on dense SIFT descriptors for every sparsity value. This makes sense because SIFT descriptors are extracted using saliency-based sampling strategy, which can capture the salient features of the image, while dense SIFT descriptors are extracted using grid-based sampling strategy, which can only capture the features of the whole scene. In the following experiments, UFLF refers to UFLF based on SIFT descriptors. Overall the performances of UFLF using L1 and L2 are comparable. For SIFT-based extraction strategy, we found the similarity measure using L1 has slightly better performance than that using L2, while the opposite conclusion is obtained for dense SIFT-based extraction strategy except for sparsity 1.0.

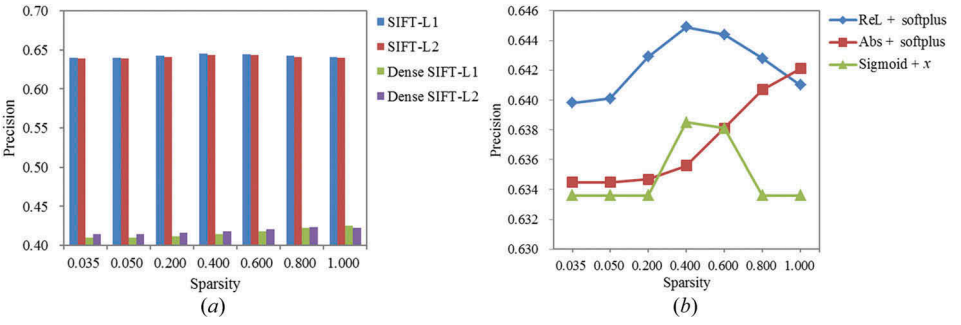

Figure 3. The average precision of UFLF over 10 classes with varying sparsity values (*a*) based on SIFT and dense SIFT and (*b*) using the three activation groups described in the text.

Three activation function groups are also considered in this study, namely 'ReL + softplus', 'sigmoid + *x*' and 'abs + softplus'. The two functions of each group are the activation functions of the hidden layer and the output layer, respectively. In Figure 3(*b*), we show the performance of UFLF using these activation function groups. Overall, the group 'ReL + softplus' achieves higher average precision over all 10 classes compared to the group 'sigmoid + *x*'. The promising results are probably because the ReL activation function can enforce the sparsity of hidden units and reduce the gradient vanishing problem. To validate this view, the function group 'abs + softplus' is also investigated for two reasons: (1) the two function groups have the same activation functions of the output layer; and (2) ReL and abs have similar function formulas.

Figure 4(*a*) indicates the best results (sparsity 0.4) for sparse features as well as the results for non-sparse features using L1 and L2 similarity measures. Non-sparse features are the learned features without using the soft threshold function to enforce more sparsity as mentioned in section 2.4. It is shown that sparse features achieve better performance than non-sparse features using the same similarity measure.

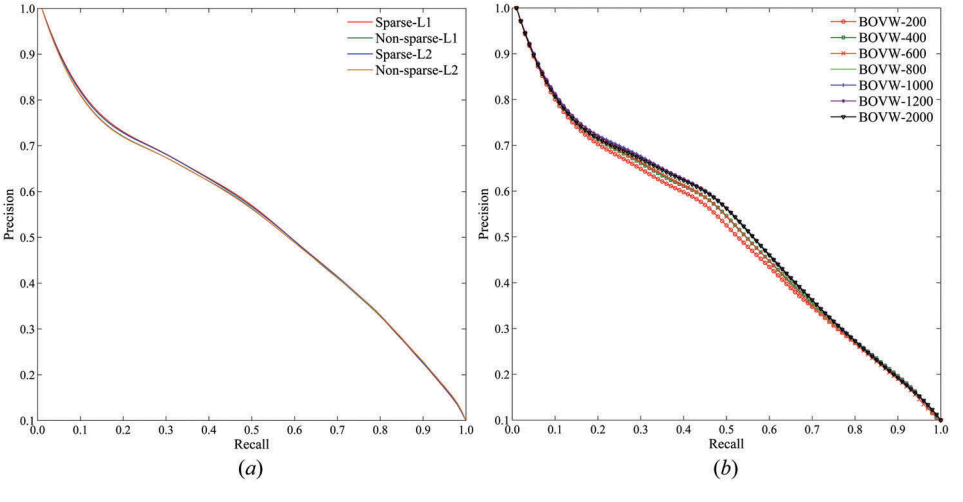

Figure 4. Graphs of precision against recall for (*a*) sparse feature against non-sparse feature and (*b*) BOVW with varying codebook size (the numbers in the legend are the codebook sizes).

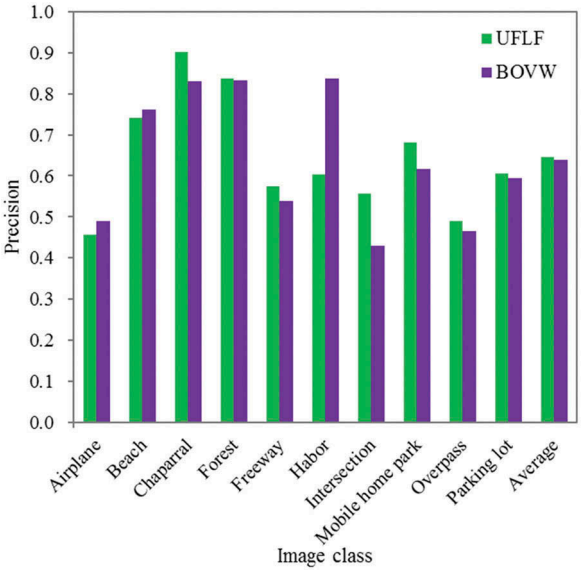

Figure 5. Comparison of the retrieval performance of UFLF and BOVW representations for different image classes.

The remainder of this section compares the performance of our UFLF to that of the BOVW representation. BOVW is a state-of-the-art method in the image retrieval literature. Figure 4(*b*) shows the performance of BOVW representation with a range of codebook sizes. It indicates that 1000 is the optimal codebook size. Figure 5 shows the performance of UFLF and BOVW for each image category. It can be seen that UFLF has better performance for most of the image categories. The last bin in the figure denotes the average precision over all image classes, and it indicates that the average precision generated by UFLF is higher than that by BOVW.

4. Conclusions

We presented a UFLF that can map low-level feature descriptors to new and sparse feature representations. Unlike previous works that focused on designing robust feature representations, UFLF can learn sparse features in an unsupervised way. We demonstrated that UFLF is more effective than BOVW using several performance metrics, and UFLF based on SIFT descriptors outperforms UFLF based on dense SIFT descriptors. We also compared three activation function groups used for UFLF to validate the advantages of ReL activation function.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by National Science & Technology Specific Projects [grant numbers 2012YQ16018505 and 2013BAH42F03], National Natural Science Foundation of China [grant number 61172174], and innovative talents project of Wuhan University [grant number 2042014kf0212].

References

Aptoula, E. 2014. "Remote Sensing Image Retrieval with Global Morphological Texture Descriptors." *IEEE Transactions on Geoscience and Remote Sensing* 52: 3023–3034. doi:10.1109/TGRS.2013.2268736.

Chen, L., W. Yang, K. Xu, and T. Xu. 2011. "Evaluation of Local Features for Scene Classification Using VHR Satellite Images." In *Proceeding Joint Urban Remote Sensing Event*, April 11–13, edited by U. Stilla, P. Gamba, C. Juergens, and D. Maktav, 385–388. Munich: IEEE.

Cheng, G., J. Han, L. Guo, Z. Liu, S. Bu, and J. Ren. 2015. "Effective and Efficient Midlevel Visual Elements-Oriented Land-Use Classification Using VHR Remote Sensing Images." *IEEE Transactions on Geoscience and Remote Sensing* 53: 4238–4249. doi:10.1109/TGRS.2015.2393857.

Cheriyadat, A. M. 2014. "Unsupervised Feature Learning for Aerial Scene Classification." *IEEE Transactions on Geoscience and Remote Sensing* 52: 439–451. doi:10.1109/TGRS.2013.2241444.

Glorot, X., A. Bordes, and Y. Bengio. 2011. "Deep Sparse Rectifier Networks." In *Proceedings of 14th International Conference on Artificial Intelligence and Statistics*, April 11–13, edited by G. Gordon, D. Dunson, and M. Dudík, 315–323. Ft. Lauderdale, FL: MIT Press.

Han, J., D. Zhang, G. Cheng, L. Guo, and J. Ren. 2015. "Object Detection in Optical Remote Sensing Images Based on Weakly Supervised Learning and High-Level Feature Learning." *IEEE Transactions on Geoscience and Remote Sensing* 53: 3325–3337. doi:10.1109/TGRS.2014.2374218.

Han, J., P. Zhou, D. Zhang, G. Cheng, L. Guo, Z. Liu, S. Bu, and J. Wu. 2014. "Efficient, Simultaneous Detection of Multi-Class Geospatial Targets Based on Visual Saliency Modeling and Discriminative Learning of Sparse Coding." *ISPRS Journal of Photogrammetry and Remote Sensing* 89: 37–48. doi:10.1016/j.isprsjprs.2013.12.011.

Hinton, G. E., and R. R. Salakhutdinov. 2006. "Reducing the Dimensionality of Data with Neural Networks." *Science* 313: 504–507. doi:10.1126/science.1127647.

Lazebnik, S., C. Schmid, and J. Ponce. 2006. "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories." In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 17–22, edited by A. Fitzgibbon, C. J. Taylor, and Y. LeCun, 2169–2178. New York: IEEE.

Lowe, D. G. 2004. "Distinctive Image Features from Scale-Invariant Keypoints." *International Journal of Computer Vision* 60: 91–110. doi:10.1023/B:VISI.0000029664.99615.94.

Newsam, S., L. Wang, S. Bhagavathy, and B. S. Manjunath. 2004. "Using Texture to Analyze and Manage Large Collections of Remote Sensed Image and Video Data." *Applied Optics* 43: 210–217. doi:10.1364/AO.43.000210.

Pati, Y. C., R. Rezaiifar, and P. S. Krishnaprasad. 1993. "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition." In *Proceeding Asilomar Conference on Signals, Systems and Computers*, November 1–3, edited by A. Singh, 40–44. Pacific Grove, CA: IEEE.

Yang, Y., and S. Newsam 2011. "Spatial Pyramid Co-Occurrence for Image Classification." In *IEEE International Conference on Computer Vision*, November 6–13, edited by D. N. Metaxas, L. Quan, A. Sanfeliu, and L. J. Van Gool, 1465–1472. Barcelona: IEEE.

Yang, Y., and S. Newsam. 2013. "Geographic Image Retrieval Using Local Invariant Features." *IEEE Transactions on Geoscience and Remote Sensing* 51: 818–832. doi:10.1109/TGRS.2012.2205158.

Zhang, F., B. Du, and L. Zhang. 2015. "Saliency-Guided Unsupervised Feature Learning for Scene Classification." *IEEE Transactions on Geoscience and Remote Sensing* 53: 2175–2184. doi:10.1109/TGRS.2014.2357078.