# *Dense Cloud Classification on Multispectral Satellite Imagery*

Kay Wohlfarth, Christian Schröer, Maximilian Klaß, Simon Hakenes, Maike Venhaus, Sönke Kauffmann, Thorsten Wilhelm, Christian Wöhler
Image Analysis Group
TU Dortmund University
44227 Dortmund, Germany
kay.wohlfarth@tu-dortmund.de

***Abstract*—In this paper we explore the capabilities of two state-of-the-art machine learning techniques, transfer learning with convolutional neural networks (CNN) and support vector machines (SVM) to distinguish between 10 cloud genera. We will evaluate these methods using images acquired by the satellite Landsat 8. The classification of cloud genera is of high general relevance for remote sensing applications such as the surveillance of atmospheric or meteorological processes. Transfer learning is of advantage because it exploits neural networks, which are known to perform well, and enables the adaption to a specific problem with relatively small training data size. We will utilize Landsat 8 images for evaluating the examined machine learning approaches because these image data are freely available in large amounts. Downscaling the Landsat 8 images utilized for training to a resolution of about 300 meters per pixel will allow for keeping the CNN and SVM size reasonably low, such that our training data set can be restricted to a moderate size.**

***Keywords*—Clouds, Classification, Machine Vision, Support Vector Machines, Convolutional Neural Network, Transfer learning, Remote Sensing, Landsat 8**

## I. INTRODUCTION

### A. *Related Work*

Automated classification of clouds from multispectral satellite imagery is an important task for a variety of meteorological applications ranging from operational and synoptic tasks to climatology. Two prominent challenges are faced when dealing with clouds in the context of remote sensing, i.e. the *detection* and *classification* of clouds. A survey of cloud detection approaches is given by [1]. A recent approach of cloud detection is presented in [2]. The cloud detection technique employed by the United States Geological Survey (USGS) is an implementation of the FMASK algorithm [3,4]. In this work, we focus on the classification of clouds which covers cloud detection, but goes beyond. The work of [5] provides an early overview of the fundamentals of pattern recognition in view of cloud classification.

Support Vector Machines (SVM) are a common tool in remote sensing and have been extensively used in the domain of ground and vegetation analysis [6]. In [7], a SVM based approach is presented to analyse cloud coverage and broader cloud structures, distinguishing between five classes on the METEO-SAT imagery. In [8], a cloud type classifier on data of the Multi-angle Imaging SpectroRadiometer (MISR) device was presented which works with reduced SVMs on a set of eight classes, consisting of five cloud types and three non-cloud classes. In [9], a comparison between Convolutional Neural Networks (CNN) and SVM classifiers is presented using Landsat 8 images for the recognition of golf courses and observed better results when using a CNN.

Cloud classification on satellite imagery combining feature extraction and neuronal networks was investigated by [10,11,12]. Advanced Very High Resolution Radiometer (AVHRR) data in polar regions was investigated by [11]. In the study, it is discriminated between five cloud genera and five land types. Twenty handcrafted features based on Gray Level Difference Vectors (GLDV) [13] and the Sum and Difference Histogram method (SADH) [14] were used. These features were fed into a Probabilistic Neural Network (PNN) [15] for learning and classification. AVHRR data in maritime regions was investigated by [12] combining 15 handcrafted features based on GLDV and SADH with a PNN classifier, distinguishing between nine cloud genera. GEOS 8 satellite images were evaluated by [10], distinguishing between six cloud classes and four land types. Singular value decomposition (SVD) and wavelet packages (WP) are used as feature extractor. PNN and Kohonen self-organized maps (SOM) [16] are employed as classifier. The generated label maps are enhanced by a Markov Random Field (MRF) [17]. In the last years, so-called Convolutional Neural Networks gained further popularity which integrate feature extraction, feature abstraction, and classification in one framework. CNNs were applied to tasks such as cloud coverage detection [18] masking out clouds [19] and the detection of overshooting tops which indicate severe weather conditions [20]. These three works [18, 19, 20] aim at detecting clouds. A newly trained CNN is presented in [21] to distinguish between five cloud classes. But no distinction of different cloud genera is given, nor it is shown how the classifier applies to full images. An overview of the mentioned classification approaches is given in Tab. 1. The classifi-cation of several cloud genera with a CNN which was trained using transfer learning has, to the best our knowledge, not been studied yet. Further, we note that our study is the first to distinguish

TABLE I. OVERVIEW OVER SELECTED PUBLICATIONS ON CLOUD CLASSIFICATION

| Publication | Dataset | | Number of Classes | | Methods | |
|---|---|---|---|---|---|---|
| | *Data Source* | *Number of Samples* | *Clodus* | *Total* | *Features* | *Classifier* |
| Welch et al. [11] | AVHRR (polar) | 1804 samples | 5 | 10 | GLDV, SADH | PNN |
| Bankert [12] | AVHRR (maritime) | 1633 samples | 9 | 10 | GLDV, SADH | PNN |
| Tian et al. [10] | GEOSAT 8 | 6 annotated imgs. | 6 | 10 | SVD, WP | PNN |
| Mazzoni et al. [8] | NASA's Terra (MISR) | 3500 labelled pixels | 5 | 8 | Handcrafted | SVM |
| Vogel [7] | METEOSAT | ≤ 1500 pixels/class | 4 | 5 | Handcrafted | SVM |
| Cai & Wang [21] | FY-2C | 200 samples/class | 5 | 5 | CNN | |
| SVM (this work) | Landsat 8 | 822 samples/class | 9 | 13 | BOF | SVM |
| CNN (this work) | Landsat 8 | 1000 samples/class | 9 | 13 | CNN | |

between nine cloud classes encompassing eight cloud genera while considering four different underground types in a variety of geographic regions.

### B. *Cloud Types of Interest*

To improve the level of detail in climatological analysis, it is desirable to distinguish between many different cloud types. Since the different species of a genus are highly ambiguous, even for the trained eye, we primarily aim at classifying the clouds of the main genera. The genera Cirrocumulus and Nimbostratus are excluded due to the lack of data. Additionally, a new class (Co) was introduced which encompasses Cumulus congestus and Cumolonimbus calvus indicating emerging thunder. When dealing with complete satellite images, extensive parts of the earth's surface might not be covered by clouds. To avoid misclassification, it is necessary to introduce additional non-cloud classes, which cover the rest of the earth's surface. In total, we have thirteen classes: Ac = Altocumulus, As = Altostratus, Cs = Cirrostratus, Ci = Cirrus, Cb = Cumolonimbus, Co = Cumulus congestus, Cu = Cumulus, Sc = Stratocumulus, St = Stratus, Wa = Water, La = Land, Ku = Coast, Sn = Snow.

## II. METHODS

### A. *Landsat 8 Data Products*

The EarthExplorer of the United States Geological Survey (USGS) offers different Landsat 8 image files acquired by two sensors: The Operation Land Imager (OLI) instrument collects image data in nine spectral bands with wavelengths between 435 nm to 2294 nm and The Thermal Infrared Sensor (TIRS) collects data for two thermal bands ranging from 1060 nm to 1251 nm. Three bands with the wavelengths 1566 nm – 1651 nm, 851 nm – 879 nm, 636 nm – 672 nm are mapped to the RGB channels and form the so called Natural Color Image. The pixel values $p_i$ are scaled such that $p_i \in [1, 255]$ with a gamma stretch of $\gamma = 2.0$. The thermal image file displays the thermal properties of a scene at wavelength 10.6 µm – 11.19 µm of the TIRS. It is also scaled such that $t_i \in [1, 255]$. Refer to the USGS Landsat 8 (2016) data [22], for details (https://landsat.usgs.gov/landsat-data-access).

### B. *Generating a Database*

We acquired 147 full resolution Landsat 8 images showing many different cloud types in regions between 60°N and 60°S. The images have widths and heights in the range between 6000 – 8000 pixel. We cut out image patches of size 320 x 320 pixel. In order to decrease the computational effort, these patches were down sampled to a size of 32 x 32 pixels. In total, 9578 small cloud image patches were collected for 13 different classes. Additionally, the corresponding patches from the thermal infrared image were stored. To achieve an equal class balance, dataset augmentation, i.e. rotation and mirroring, was performed which finally yields to 1000 samples per class. 90 % are used as training set and 10 % are used as an independent test set for performance evaluation. The labeling procedure of cloud patches involved five people and took a few sessions of collaborative labeling.

### C. *Transfer Learning with CNNs for Cloud Classification*

CNNs are a state-of-the-art technique for image classification. For a detailed synopsis on deep learning in neural networks and a summary of recent advances on CNNs see for instance [23,24,25]. A CNN for classification can be seen as a pipeline where an input image is fed into the neural network and a probability of class membership for this image is generated as output. Within, the CNN is organized in layers where each layer implements a certain operation on the output of the preceding layer. So-called convolutional layers implement learnable filter banks which are used to extract features from the input. Modern CNNs are usually trained with algorithms derived from stochastic gradient descent (SGD) [25]. A common technique in machine learning is the so-called transfer learning in which a previously trained neural network is adopted to a new problem. As stated in [25], "Transfer learning and domain adaptation refer to the situation where what has been learned in one setting is exploited to improve generalization in another setting". The advantage is that one does not have to fully train a network from scratch and one can exploit an already existing architecture. This is especially useful when the training data

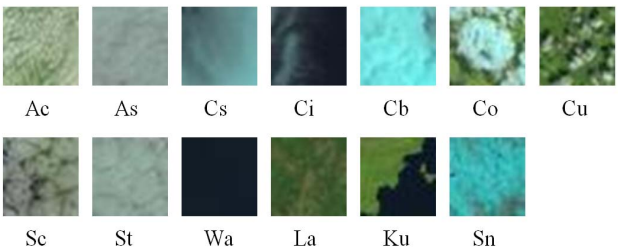


Fig. 1. Examples of 32 x 32 image patches of the 13 classes

size is small compared to the dataset used for the initial training of the model. We employ transfer learning by starting with the well recognized convolutional neural network termed Alex-Net [26] and adapt it to the problem size by pruning several layers until we arrive at an architecture we denote as cloudnet. AlexNet was introduced by Krizhevsky et al. [26] to classify 1000 different classes and reached top scores for the ImageNet LSVRC-2012 contest [27]. It was designed for large problems with hundreds or even thousands of classes and sample numbers in the range of $10^5$-$10^6$. Since the task of cloud classification with the given database requires a less representational model capacity than provided by the AlexNet, we prune the middle part of the AlexNet preserving only the first two convolutional layers and pooling layers. An untrained third convolutional layer and the untrained fully connected part are added. Our cloudnet is then trained with the AdaGrad solver for 10.000 training iterations which takes roughly ten minutes on an Intel i5 processor. A scheme of the cloudnet architecture can be seen in Fig. 2.

### *D. Including Thermal Information*

In order to incorporate thermal information, two experiments were carried out: At first, the cloudnet is trained from scratch with 4D-data consisting of the three color channels and the infrared channel. Therefore, the whole dataset is expanded by adding a fourth IR-channel to the three color channels which was concurrently retrieved during the labeling procedure. In the case of 4D-input, the pretrained layers of the AlexNet can not be re-used for our cloudnet, because they were designed for three channels only. We expand our cloudnet by a fourth channel and train it from scratch with the newly generated dataset. In the second experiment, the red channel is replaced with the IR-channel and the information of the red channel is mixed into the blue and green channel (30% red + 70% green and 30% red + 70% blue). The generated dataset has three channels only and can be readily fed into our cloudnet reusing the AlexNet kernels.

### *E. SVM for Cloud Classification*

The concept of the Support Vector Machine (SVM) is based on statistical learning theory and has been introduced by [28]. The aim is to find a hyperplane in the feature-space, i.e. a decision boundary which separates between two classes. Support vectors are those points located closest to the boundary between the classes and are used to define the hyperplane. If linear separation of the given data is not possible, soft Margin Classification or the application of a kernel may be employed. Refer to [28] and [29] for a detailed description of the SVM. We used Matlab for the implementation of the SVM. The framework works with grayscale images. To preserve color and infrared information, we place the channels next to each other which yields a single

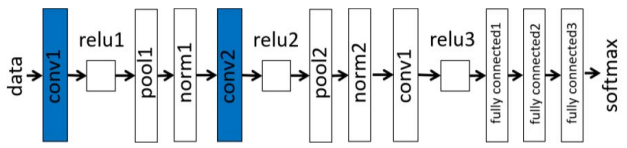


Fig. 2. Architecture of the cloudnet, pretrained kernels are blue

channel image. For color-only classification, the three color channels form 96 x 32 images. Including infrared information generates 64 x 64 images. In total, 822 training samples were used for each class. We extract features, which are then used to train a multiclass SVM. We acquire features for the multiclass SVM according to the bag of keypoints approach introduced by Csurka et al. [30]. They use Scale-invariant feature transform (SIFT) descriptors [31,32] because of their invariance to image transformation and lighting variation. Speeded up robust features (SURF) were employed, as Bay et al. [33] showed that they outperform SIFT for classification tasks, while still being faster to compute. Following [30], the set of quantized feature vectors defined by the cluster centers are referred to as keypoints. With the established bags of keypoints we can now train the actual classifier. At first, the patch descriptors are assigned to the vocabulary by searching the nearest neighbor for a given point in feature space. Counting points assigned to the cluster centers yields a histogram, which now becomes the feature vector as an input for the SVM classifier [30]. Our classifier relies on an error-correcting output code model (ECOC) that reduces a multiclass classification problem to a combination of binary problems [34]. We employed a one-versus-one design leading to 78 binary classifiers. Decoding, as described by [35], involves applying the binary classifiers, which yields a codeword for each data sample. The class is obtained by comparing this codeword to the base codewords defined in the coding matrix. Depending on the decoding scheme, the closest codeword is determined, which corresponds to the class label. The training takes appx. 30 – 60 min.

### *F. Window Classifier*

The two classifiers, CNN and SVM, were designed to handle small images that show a single class. For several applications, it is favorable to create a map, which indicates the cloud type that is most likely located at a given coordinate in a large image. This is done by adopting a window classifier which scans the image block-wise: For every pixel, the surrounding 32 x 32 block is taken as an input for the classifier. As for the image patches, we down-sample the full image by a factor of ten to match the scale of the training data. To speed up the map generation, the window classifier has an option for subsampling. Fully Convolutional Networks (FCN) are known to generate good densely annotated maps in remote sensing images [36]. The FCN segments an image in one shot and needs large amounts of densly annotated training data. The training procedure also involves transfer learning and finetuning to the segmentation task [37]. But the generation of the required data is even more time consuming compared to the collection of small image patches. Additionally, experience from the labeling sessions showed that densely annotating complex cloud formations is marked by high ambiguity at borders between cloud types which is likely to negatively affect the classification performance.

### *G. Postprocessing using Markov Random Fields*

Applying the window-classifier yields a dense label map indicating the predicted class for each pixel. Such label maps can exhibit classification noise as a consequence of

misclassification, for example at transitions between classes. Markov Random Fields (MRF) utilize the confidence of the classificator and the spatial context of a single classified pixel. MRF, as described by Geman & Geman (1984) were used by Tian et al. (1999) in the context of cloud classification and lead to an increased classification accuracy. Therefore, we adopt the same approach. Applying MRF requires the (pseudo-) probabilities of the classifier which are only present for the CNN. Thus, MRFs are applied to the CNN results only.

## III. Results

### A. *Classification Accuracy CNN*

The training procedure yields an overall accuracy of 85.8% on all thirteen classes measured on the test set. Significant misclassification of Altostratus and Altocumulus clouds affect the overall performance. Considering the confusion matrix of this experiment (see Fig. 3), several effects can be identified: Most cloud types are classified very well with an accuracy of more than 90%. In 16% of cases, Cu is misclassified as La. Some Cumulus clouds can be very small, meaning they are just slightly silhouetted against the background and are therefore confused. Because of the gradual transition between Sc and Cu clouds, we observe 33% of cases, in which Sc clouds are mistakenly classified as Cu. As and Ac are misclassified as St in a significant number of cases. This is due to their similar textural appearance (see Fig. 1). Based on the lack of unique textural attributes and its close resemblance to Cu, St and Sc, Ac is mixed up with these types in a sign ificant number of cases.

### B. *Using Thermal Information*

Two experiments were carried using the thermal channel when training the CNN: When training the cloud net architecture from scratch with 4D-data, the accuracy on 13 classes is less than 80% after the training curve saturated. This result indicates that training from scratch with four channels is not as accurate as employing transfer learning with only three channels. In the second experiment, the infrared information was incorporated into the three channels: Finetuning the cloudnet, which can make use of the pretrained AlexNet kernels, yields an overall accuracy of 86.1%. This is 0.3 % higher than for the pure color channels.

### C. *Classification Accuracy SVM*

In the first step, we employ the three channel color data for training the SVM. The overall accuracy on the test set is 77.4%. In the second step, the infrared channel is added. The confusion matrix for the full test-set of 13 classes containing 100 images per category is shown in Figure. 4. The results of the SVM-classification are now very accurate (95.4%). Significant misclassification can only be noted for Cumulus, Stratus, and Stratocumulus clouds. Both Cumulus and Stratus classification suffers from the similarity to Altocumulus and Altostratus, respectively.

### D. *Generating a Dense Cloud Map*

To visualize the results, we chose an image which exhibits complex patterns of Cumulus congestus, thunder and combinations of Cu and Sc (Thunder Fig. 5). The original image (7631 x 7781 pixel) was down-sampled by a factor of 10. The moving window is implemented with an adaptable step size. Here, a step size of two is chosen for both directions which means that one out of four pixels is considered. Thus, appx. 145,000 calls of the classifier are necessary to generate a label map of one of the images. This procedure takes roughly half an hour to forty minutes on an Intel i5 processor. For step size 8, a coarse classification of the image can be obtained in less than one minute. As it can be seen in Fig. 5, the vast majority the surface types are identified correctly. The common cloud types such as Cumulus, Stratocumulus, and Stratus clouds are identified with high accuracy as well. However, there seems to be a slight ambiguity whether a given cloud formation is an accumulation of Cumulus clouds or already an area covered in Stratocumulus clouds. This indicates the smooth transition between these two classes. Cumulus congestus clouds are detected reliably making it easy to find emerging and fully developed thunderstorms. The prominent overshooting tops of the Cumolonimbus clouds are detected with high accuracy and even the Cirrostratus extensions are found. The window classifier precisely segments the different clouds and underground types in the examples. From visual inspection, it

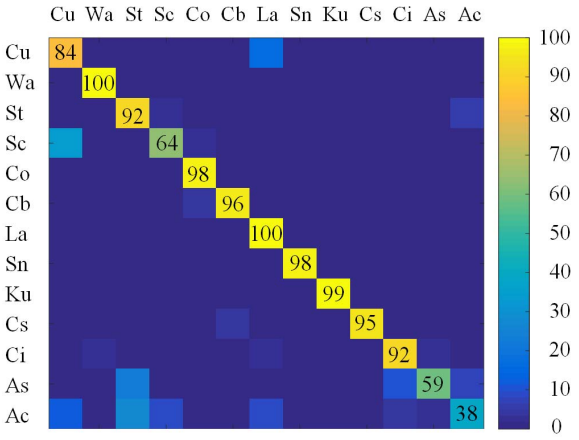


Fig. 3. Confusion matrix of cloudnet using transfer learning with 13 classes and three channels. Values in percent. Overall accuracy: 85.8 %

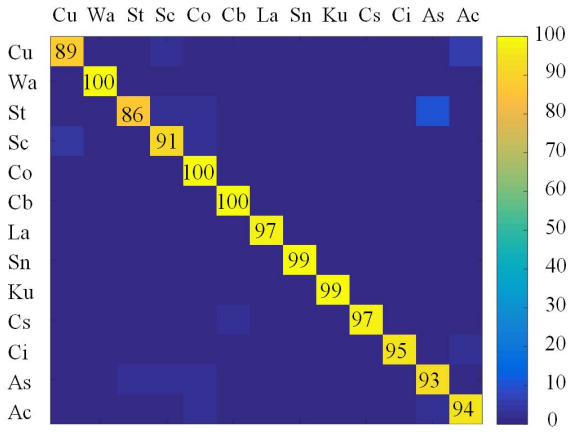


Fig. 4. Confusion matrix of SVM with 13 classes and four channels. Values in percent. Overall accuracy: 95.4 %

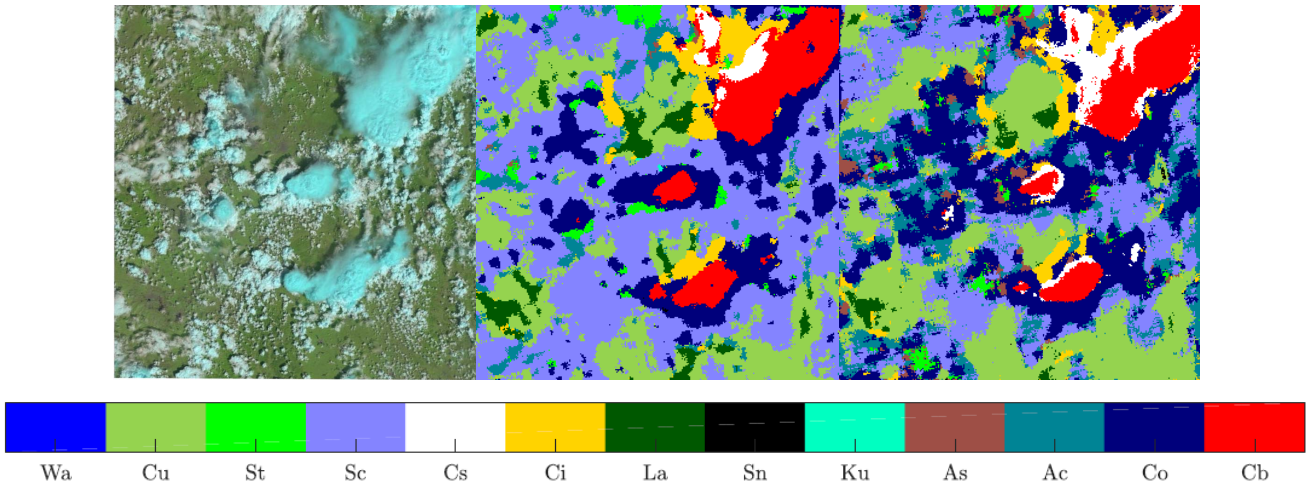


Fig. 5. Window classifier applied to thunder cells (Xānxī province). Left: Original image, Middle: Label map CNN, Right: Label map SVM

becomes clear that the label maps using the CNN classifier and the SVM are similar in most of the cases. However, there are slight differences: For example, the SVM tends to find more Cu congestus in Fig. 5 while the CNN sees these clouds as Sc.

A comparison against a densely annotated ground truth is of limited expressiveness here, since it is not always possible, even for the trained eye, to separate between adjacent clouds. Experience from the labeling procedure showed that densely annotating complex cloud forms (Fig 2.) will yield high ambiguities and thus we rely on testing with the independent test set for performance evaluation (Sec. III A – III C).

### *E. Postprocessing using MRF*

Applying MRF postprocessing yields a label map with reduced classification noise (see Fig. 6). Note that this approach is only suitable for the CNN, since we do not obtain confidence or pseudoprobabilities for class membership from the SVM's.

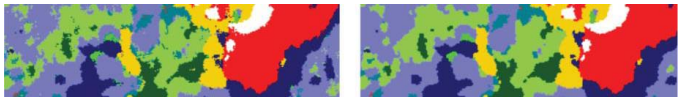

Fig. 6. Left before, right after MRF postprocessing (Thunder in Xānxī province)

## IV. Discussion

Training the SVM with three color channels yields an overall accuracy of 77.4%. Using four channels (color + thermal) for training, the SVM yields the highest overall accuracy of 95.4%. This indicates, that the thermal channel adds discriminative information such that the classification results of the SVM become better. Using color only, the transfer-learned cloudnet has an accuracy of 85.8%. If the thermal channel is integrated into the three channels, the overall accuracy is 86.1% and thus slightly larger. If we use four channels (color + thermal) to train the cloudnet from scratch, the accuracy is below 80%. This may show, that transfer learning works better than training from scratch, even though the inclusion of a fourth channel should have added more information to the dataset. One reason that the classification of the SVM is higher can be, that the statistical feature extractors used in the SVM are better suited for the problem compared to the learned kernels of the CNN.

The labeling procedure took the longest which is several sessions of collaborative labeling involving up to five people. Nevertheless, training data is necessary for any supervised training approach. Since the dataset is still small in the sense of deep learning, any deep learning architecture will be oversized. Thus, we introduced the cloudnet (pruned AlexNet) and performed transfer learning which exploits the previously learned general features. The current implementation of the CNN is much faster compared to the SVM. Training the CNN takes roughly ten minutes while training the SVM takes more than half an hour on a CPU of a regular PC working station (Intel i5 CPU). Both classifiers show consistent results when being used for densely annotating the example images. Using the MRF post-processing, classification noise of the CNN can be reduced.

## V. Conclusion

We used state of the art SVM and CNN classification methods to classify 13 different land and cloud categories captured in Landsat 8 imagery. We found that both approaches perform well on most classes. The accuracy of our cloudnet architecture trained from scratch for 4D input was lower compared to the cloudnet which used transfer learning to predict 3D input. The higher recognition rates in the second case indicate that transfer learning can be regarded as a suitable technique reusing and adapting a previously trained network to the problem of cloud classification. For the SVM, thermal information is essential leading to the highest recognition rates on all thirteen classes. For dense image analysis, a window classifier was implemented, which annotates not only regions of interest but the whole image frame by pixel-wise prediction. The window classifier produces consistent classification results even for complex

thunder cells. The results of the CNN can be further refined by applying a MRF postprocessing.

## REFERENCES


[1] Chandran, J., "A survey of cloud detection techniques for satellite images," International Research Journal of Engineering and Technology IRJET, vol. 2, no. 9, pp. 2485–2490, 2015.

[2] Xie, F., Shi, M., Shi, Z., Yin, J. and Zhao, D., "Multilevel cloud detection in remote sensing images based on deep learning," IEEE JSTARS, vol. 10, no. 8, pp. 3631–3640, 2017.

[3] Zhu, Z. and Woodcock, C.E., Object-based cloud and cloud shadow detection in Landsat imagery, Remote Sensing of Environment, vol. 118, pp 83–94, 2012.

[4] Zhu, Z. and Woodcock, C.E., Improvement and Expansion of the Fmask Algorithm: Cloud, Cloud Shadow, and Snow detection for Landsats 4-7, 8, and Sentinel2 images, Remote Sensing of Environment, vol. 159. 2014.

[5] Pankiewicz, G.S., "Pattern recognition techniques for the identification of cloud and cloud systems," Meteorological Appl., vol. 2, no. 3, pp. 257–271, 1995.

[6] Mountrakis, G., Im, J. and Ogole, C., "Support vector machines in remote sensing: A review," fISPRSg Journal of Photogrammetry and Remote Sensing, vol. 66, no. 3, pp. 247 – 259, 2011.

[7] Vogel, R., "Development of an automated cloud detection and cloud classification algorithm using support vector machines applied to meteosat seviri data for the area of germany," PhD dissertation, Universitt Bonn, 2011.

[8] Mazzoni, D., Horva´th, A., Garay, M. J. , Tang, B. and Davies, R., "A misr cloud-type classifier using reduced support vector machines," in Eighth Workshop on Mining Scientific, 2005, p. 19.

[9] Ishii, T., Nakamura, R., Nakada, H., Mochizuki, Y. and Ishikawa, H., "Surface object recognition with cnn and svm in landsat 8 images," in Machine Vision Applications (MVA), 2015 14th IAPR International Conference on. IEEE, 2015, pp. 341–344.

[10] Tian, B., Shaikh, M. A., Azimi-Sadjadi, M. R., Haar, T. H. V., and Reinke, D. L., "A study of cloud classification with neural networks using spectral and textural features," IEEE Transactions on Neural Networks, vol. 10, no. 1, pp. 138–151, Jan 1999.

[11] Welch, R., Sengupta, S., Goroch, A., Rabindra, P., Rangaraj, N. and Navar, M., "Polar cloud and surface classification using avhrr imagery: An intercomparison of methods," Journal of Applied Meteorology, vol. 31, no. 5, pp. 405–420, 1992.

[12] Bankert, R. L., "Cloud classification of avhrr imagery in maritime regions using a probabilistic neural network," Journal of Applied Meteorology, vol. 33, no. 8, pp. 909–918.

[13] Weszka, J. S., Dyer, C. R. and Rosenfeld, A. "A comparative study of texture measures for terrain classification," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-6, no. 4, pp. 269–285, April 1976.

[14] Unser, M., "Sum and difference histograms for texture classification," IEEE Transactions on PAMI, no. 1, pp. 118–125, 1986.

[15] Specht, D. F., "Probabilistic neural networks," Neural networks, vol. 3, no. 1, pp. 109–118, 1990.

[16] Kohonen, T., "The self-organizing map," Proceedings of the IEEE,vol. 78, no. 9, pp. 1464–1480, Sep 1990.

[17] Geman, S. and Geman, D., "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-6, no. 6, pp. 721–741, Nov 1984.

[18] Le Goff, M., Tourneret, J.-Y., Wendt, H., Ortner, M., Spigai, M., Deep Learning for Cloud Detection. 10 (6.)-10 (6.). 10.1049/cp.2017.0139, 2017.

[19] Mateo-Garcia, G., Gomez-Chove, L. and Camps-Valls, G., Convolutional neural networks for multispectral image cloud masking, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2017.

[20] Kim, M., Lee, J. and Im, J., Deep learning-based monitoring of overshooting cloud tops from geostationary satellite data, GIScience and remote sensing, pp. 1-30, 2018.

[21] Cai, K. and Wang, H., Cloud classification of satellite image based on convolutional neural networks, 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2017.

[22] D.o.t.I. USGS, Landsat 8 (L8) Data User Handbook, version 2.0 ed., March 2016. Available at USGS under following link: http://landsat.usgs.gov/documents/Landsat8DataUsersHandbook.pdf

[23] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X. and Wang, G., "Recent advances in convolutional neural networks," CoRR, vol. abs/1512.07108, 2015.

[24] Schmidhuber, J., "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85 – 117, 2015.

[25] Goodfellow, I., Bengio, Y. and Courville, A., Deep Learning. MIT Press, 2016, http://www.deeplearningbook.org.

[26] Krizhevsky, A., Sutskever, I. and Hinton, G.E., "Imagenet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[27] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A and Fei-Fei, L., "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision (IJCV), vol. 115, no. 3, pp. 211–252, 2015.

[28] Cortes, C. and Vapnik, V., "Support-vector networks," Machine learning, vol. 20, no. 3, pp. 273–297, 1995.

[29] Cristianini, N. and Shawe-Taylor, J., An introduction to support vector machines and other kernel-based learning methods. Cambridge university press, 2000.

[30] Csurka, G., Dance, C., Fan, L., Willamowski, J. and Bray, C., "Visual categorization with bags of keypoints," in Workshop on statistical learning in computer vision, ECCV, ser. 1, no. 22. Prague, 2004, pp. 1–2

[31] Lowe, D. G., "Object recognition from local scale-invariant features," in Computer vision, 1999. The proceedings of the seventh IEEE international conference on, vol. 2. IEEE, 1999, pp. 1150–1157.

[32] Ke, Y., Sukthankar, R., "Pca-shift: A more distinctive representation for local image descriptors", Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 506, 2004.

[33] Bay, H., Ess, A., Tuytelaars, T. and Van Gool, L., "Speeded-up robust features (surf)," Computer vision and image understanding, vol. 110, no. 3, pp. 346–359, 2008.

[34] Dietterich, T. G. and Bakiri, G., "Solving multiclass learning problems via error-correcting output codes," Journal of artificial intelligence research, vol. 2, pp. 263–286, 1995.

[35] Escalera, S., Pujol, O. and Radeva, P., "On the decoding process in ternary error-correcting output codes," IEEE Transactions on PAMI, vol. 32, no. 1, pp. 120–134, 2010.

[36] Sherrah, J., Fully Convolutional Networks for Dense Semantic Labelling of High Resolution Aerial Imagery, CoRR, abs/1606.02585, 2016.

[37] Shelhamer, E., Long, J. and Darrell, T., Fully Convolutional Networks for Semantic Segmentation, CoRR, abs/1605.06211, 2016.