Proceedings of the 6th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS'20)
~~Prague, Czech Republic~~ Virtual Conference – August, 2020
Paper No. MVML 107
DOI: 10.11159/mvml20.107

Taxonomy of Modelling Strategies for Handling Imbalanced Datasets

Iren Valova[1], Christopher Harris[2], Natacha Gueorguieva[2]
[1]Computer and Information Science Department, University of Massachusetts Dartmouth
285 Old Westport Rd, Dartmouth, MA, USA
Iren.Valova@umassd.edu
[2]Department of Computer Science, College of Staten Island/City University of New York
2800 Victory Blvd, NY, USA
Christopher.Harris@cix.csi.cuny.edu; Natacha.Gueorguieva@csi.cuny.edu

***Abstract* –** Class imbalanced datasets which represent real world problems are a challenge for training deep learning neural networks as they are designed to handle balanced class distributions. CNN-based topologies for image classification consider class balanced data or very low level imbalanced one for training and perform poorly when the dataset does not satisfy these conditions. This continuous problem is tackled with different strategies and methods but the ones that generate artificial data to achieve a balanced class distribution are more versatile than modifications to the classification algorithm and cost-sensitive approaches. In this paper we propose a taxonomy of modelling strategies for handling imbalanced datasets based on current approaches and algorithms. We also extend our recently proposed topology In-Between Layers Modular (IBLM) Residual Neural Network which widens the convolutional layers by adding feature planes interpreted as increase of filter numbers for each convolution layer of our residual module, and adds some topology changes. We demonstrate the IBLM ResNet classification performance on imbalanced dataset using data level preprocessing algorithms, techniques and new ensembles (Borderline SMOTE and K-Means SMOTE).

***Keywords*:** Imbalanced Dataset, Image Classification, Convolutional Neural Networks, Ensemble

1. Introduction

Deep Learning Neural Networks (DLNNs) proposed in recent years have enabled massive performance gains in tasks of various fields including computer vision, biology, finance, natural language processing etc. The ability of Deep Learning Algorithms (DLA) to automatically learn significant features has become increasingly important as the amount of data and range of applications for machine learning methods continues to grow. These algorithms represent DLNN architectures which are designed to mimic the function of the human cerebral cortex and learn complex mappings by transforming the inputs through multiple layers of nonlinear processing. DLNNs feature hierarchy increases complexity and abstraction which makes them capable of handling very large, high-dimensional data sets with millions of parameters.

With the development in Deep Learning (DL), Convolutional Neural Networks (CNNs) achieve better performance and in a lot of cases, produce human-competitive results when compared with conventional image classification methods [1]. They are considered as one of the most popular DL models due to less prior knowledge input needed while providing an end-to-end learning framework. The idea of CNNs is to create a neural network which learns by utilizing different filters (kernels) to detect various types of features in an image. Each convolutional layer summarizes the input by extracting features of interest from it and produces feature maps in response to different feature detectors. Each layer builds on the output of the previous one, so the model gathers information to achieve a bigger picture of the image and therefore produces a higher-level feature detection [2].

The architecture of a CNN is similar to that of the connectivity pattern of neurons in the human brain and has been inspired by the organization of its Primary Visual Cortex (PVC). The latter has three main properties: a) two dimensional special map organization which imitates the structure of the image in the retina; b) simple cell's activity concentrated in spatially localized receptive fields; c) complex cell's actions which are invariant to small shifts in the position of the respective features. Similarly to the above brain properties, CNNs are a specialized kind of neural networks for processing data with a grid-like topology, detector units designed to simulate the properties of the PVC simple cells, as well as different pooling strategies which mimic the brain complex cells [1, 3].

The convolution improves machine learning system in general because of its integrated capabilities: sparse interactions, sharing, and equivalent representations (translation invariance in image recognition framework). Sparse interaction is accomplished by using kernels (much smaller than the input) which occupy only a small fraction of image pixels. Sharing of learned features by all neurons in a layer in the form of strides through the entire image leads to parameter sharing. The achievement of translation invariance originates from the identification of the learned features which is independent from their location [4].

2. Imbalanced Datasets

Imbalanced learning focuses on how an intelligent system can learn when it is provided with imbalanced data. Solving imbalanced learning problems is critical in various data-intensive NN systems, as surveillance, security, Internet, finance, biomedical, defence, computer vision and more. Due to the inherent complex characteristics of imbalanced data sets, learning from such data requires new understandings, principles, algorithms, and tools to transform the given dataset efficiently into information and knowledge representation. The main issue when addressing such a learning problem appear when the accuracy achieved for each class vary significantly. Therefore, the obvious complications affect the effectiveness of accuracy as well as error rate in determining the performance of the classifiers [5].

Various research works demonstrate that the distribution of training data has a significant impact on the performance of DLNNs and in particular on the CNNs. This situation occurs because the learning process of most classification algorithm is often biased toward the majority class examples, so that minority ones are not well modelled into the final system. The CNN classifiers using balanced datasets achieve more accurate performance when compared to imbalanced one. However, considering that CNNs originally are not designed to handle imbalanced datasets, such cases have a severely negative impact on their overall performance [6, 7].

We propose taxonomy of modelling strategies for handling imbalanced domains shown in Fig. 1 which includes some recently developed approaches. As main strategies we consider Data Level Preprocessing, Algorithm Modification and Cost-Sensitive Learning.

Data Level Preprocessing approaches can be grouped into the following four main categories: a) resampling (over-resampling and under-resampling; b) advanced resampling; c) hybrid sampling; d) augmentation.

Initial implementations of the first category (random under-sampling and random-oversampling) are based on modification of imbalanced classes by copying or eliminating randomly chosen existing class instances. The latter does not lead to exactly balanced distribution, but rather to a distribution that is better handled by the classifier. This issue is addressed by the Synthetic Minority Over-sampling Technique (SMOTE) [8]. It alters the training and validation datasets by adding new synthetically generated minority class instances through extrapolation, causing the class distribution to become more balanced. In category (b) we include all approaches based on extension of SMOTE functionality as Borderline SMOTE [9], K-Means and SMOTE [10], Density-Based DBSMOTE [11], Majority Weighted Minority Oversampling Technique (MWMOTE) [12] as well as algorithms and techniques based on clustering [13], ensemble sampling [14], evolutionary undersampling [15]. Hybrid methods category (c) combines oversampling and undersampling, so that the dataset balance is achieved by neither losing too much information, nor suffering from overfitting. Two examples of hybrid techniques that have been developed include SMOTE+Tomek and SMOTE+ENN [12], where the first one oversamples the minority class, while the second one undersamples the majority class. Category augmentation technique (d) alters images in order to create new images that are like the original one, but have been transformed. Data augmentation could involve axis flipping, change in brightness range, zoom level modification, rotation, and application of other pre-processing functions [16].

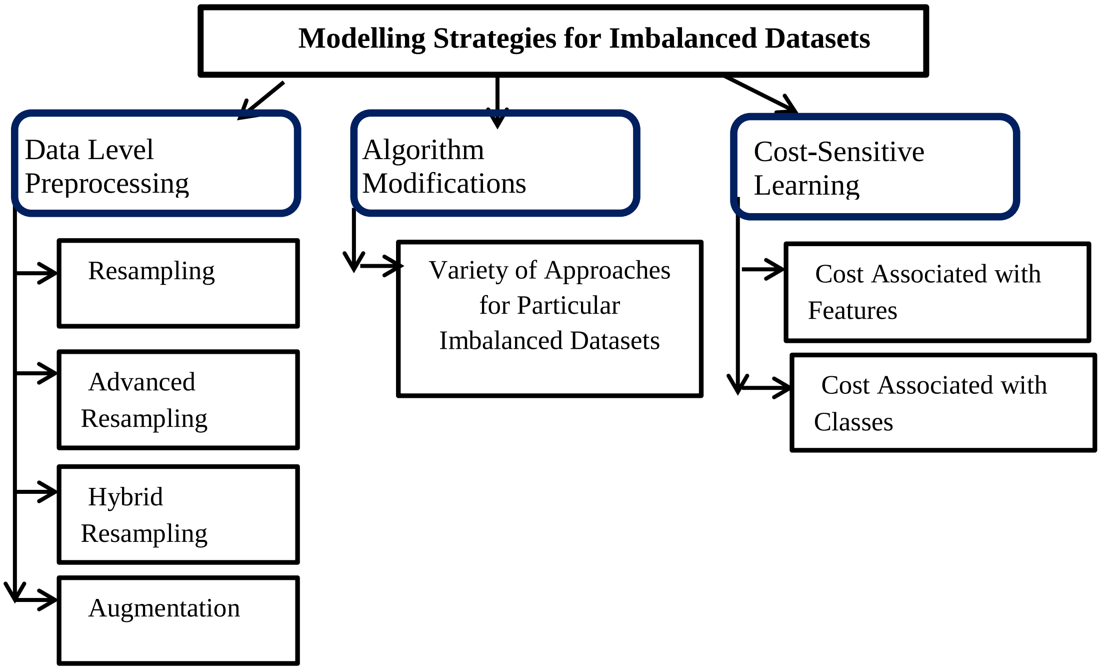

Modelling Strategies for Imbalanced Datasets
Data Level Preprocessing
Algorithm Modifications
Cost-Sensitive Learning
Resampling
Advanced Resampling
Hybrid Resampling
Augmentation
Variety of Approaches for Particular Imbalanced Datasets
Cost Associated with Features
Cost Associated with Classes

Fig.1: Taxonomy of imbalanced datasets strategies.

Algorithm Modifications are oriented towards the adaptation of respective learning method to the class imbalance issues. They present an alternative approach to the *Data Level Preprocessing* strategy for handling imbalanced datasets. Instead of focusing on modifying the training and validation datasets in order to resolve their skewed distributions, this approach aims at modifying the classifier learning procedure itself. Different specific approaches presented so far in [17, 18, 19] address particular datasets used in each research but do not offer the flexibility for more universal implementation.

Cost-Sensitive Learning combines approaches involving both, the *Data Level Preprocessing* and *Algorithm Modifications*, considering minimization of the higher costs for the misclassification of patterns of all dataset classes. There are two distinctive opinions concerning cost-sensitive classifiers which divide them into two types: a) cost associated with features; b) cost associated with classes. The first type assumes that obtaining a certain feature comes with a given cost. This determines the evaluation procedure and aims at creating a classifier that obtains the best possible predictive performance, while utilizing features that can be obtained at lowest possible cost (or the sum of costs being below a given threshold). The second type assumes costs associated with some specific classes having high-cost samples. Therefore making errors on these classes increases the classifier cost [12].

3. Experiments

The dataset discussed in this work is Intel Image Classification dataset which consists of six classes. (https://www.kaggle.com/puneet6060/intel-image-classification/version/2). The dataset represents different natural scenes, sample images of which are shown in Fig. 2. As constructed, the dataset is imbalanced, as shown in Fig. 3. While the imbalance is not extreme, it does still present a problem for model training.

3.1. Experimental Settings

Dataset. The dataset consists of 6 classes of total size 17,034 images split into a training (11920 samples), validation (3404 samples), and testing (1710 samples) sets in the ratio 70-20-10. Although validation dataset is ignored in many DLNNs works, we consider its importance as an unbiased evaluation of the model that is a fit on the training set while tuning hyperparameters and allowing visualization of training performance. The validation subset is a crucial part of training the model to avoid overfitting and underfitting which improves the test classification results of the chosen topology. Besides this, validation accuracy can be used to determine when to perform certain actions during training. For example callback function such as EarlyStopping or ModelCheckpoint in Keras can stop the training process when the model is stabilized, and no more epochs contribute to the model learning as well as to save the weights of the model at its peak. To handle the class imbalance, the training and validation datasets are resampled using SMOTE [8], Borderline SMOTE [9] and K-Means SMOTE [10].

Fig. 2: Samples from Intel Image Classification Dataset.

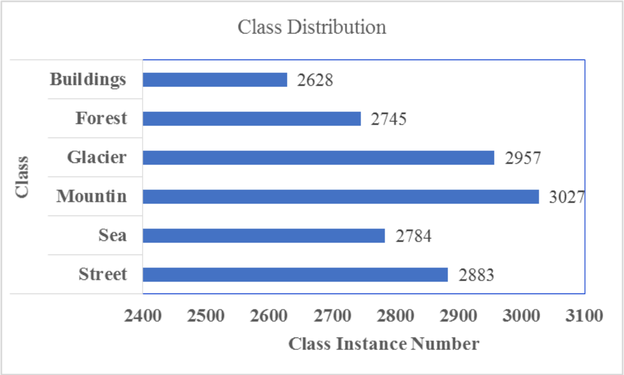


Fig. 3: Dataset class distribution.

Performance Evaluation. We present the test results using various metrics, including precision and recall, confusion matrices, F-1 score, ROC Curve and AUC. Increasing the performance of the proposed models is done on building two styles of ensemble and evaluation of their performance [20].

Implementation. We use the following three model architectures in our experiments: In-Between Layers Modular (IBLM) [21], Wide Residual Network (WRN), and traditional feed-forward CNN. Both IBLM and WRN have a depth of 10 layers and width factor of k= 2. For all experiments we use Python 3.7, Tensorflow and Keras frameworks and related libraries; ReLU activation function and Adam as an optimizer, with learning rate $\eta = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and epsilon = 1e-7, batch size of 32, dropout rate set at 0.5 and 50 epochs for training. We train all models on one machine with Nvidia GeForce GTX 1650 Super GPU. For training the CNN, epochs took approximately 40 seconds; for IBLM and WRN, about 125 seconds.

3.2. Experimental Results

Training and validation accuracy for IBLM [21], WRN and CNN for No Resampling, SMOTE, Borderline SMOTE and K-Means SMOTE are shown in Fig. 4, Fig. 5 and Fig. 6.

Analysis of IBLM (Fig.4): Borderline SMOTE model is more accurate than the others, but the difference between it and the worst model, No Resampling, is about 0.05 for each epoch. This is reflected in the train loss curves as well. On the validation set, all IBLM models experience oscillations for accuracy and loss. SMOTE and K-Means SMOTE models have the least variance in the spikes for the validation set, whereas Borderline SMOTE and No Resampling change values more sharply. Training and validation losses follow the same pattern.

Analysis of WRN (Fig. 5): For training set, K-Means SMOTE model has the best performance and No Resampling preforms the worst (Fig. 5). This is also seen in the training loss curves. For the validation set, all WRN models experience oscillations. The Borderline SMOTE model has the most variance in validation accuracy values, with K-Means being the most stable. Until epoch 25, the general trend for all WRN models is increasing the accuracy which after epoch 25 starts to decrease. The validation loss curves for all WRN models follow the same pattern.

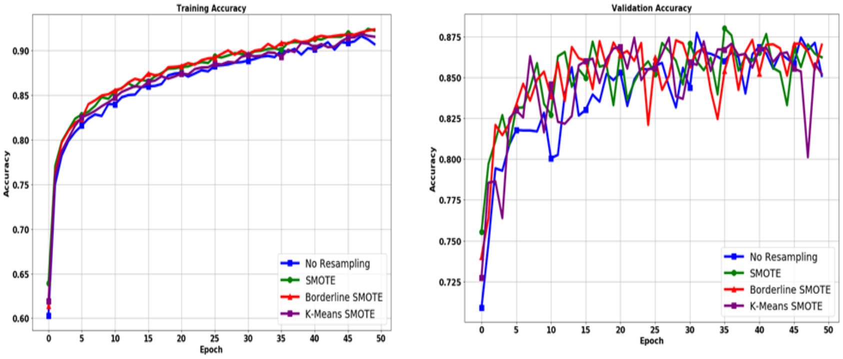


Fig. 4: Training and validation accuracy for IBLM.

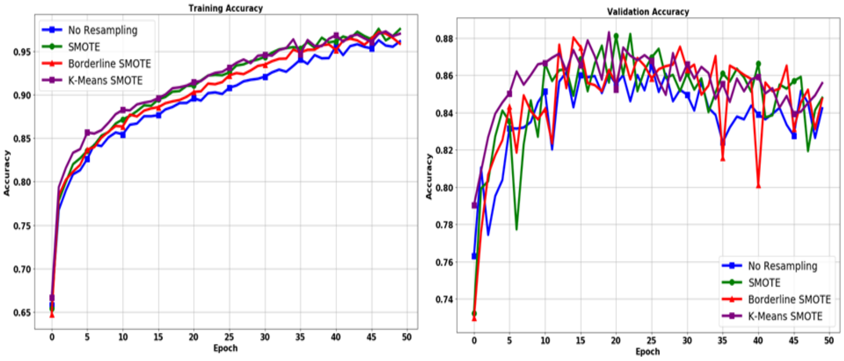


Fig. 5: Training and validation accuracy for WRN.

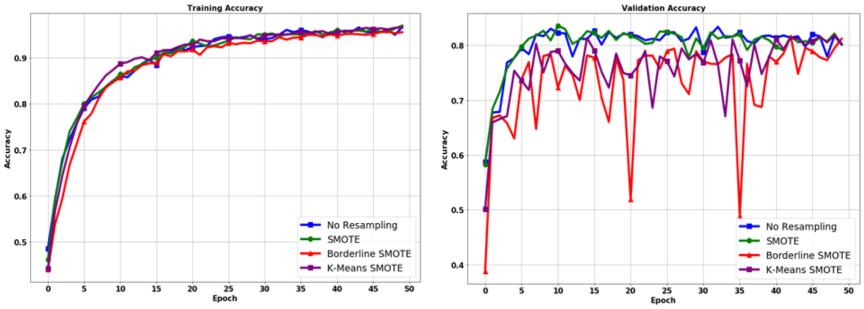


Fig. 6: CNN Training and Validation Results.

Analysis of CNN (Fig. 6): Training accuracy for SMOTE and No Resampling CNN models have an almost identical curves. Validation set accuracy for K-Means SMOTE and specifically for Borderline SMOTE models have large oscillations. Validation losses for Borderline SMOTE have the highest loss value, followed by K-Means SMOTE.

3.3. Performance Results

This section describes the evaluation of our models based on the test set for IBLM, WRN and CNN. Values reported in Table 1 are the macro average of the metric. Comparing performance testing results of all models with different approach of resampling starting with No Resample through SMOTE, Borderline SMOTE and K-Means SMOTE.

Table 1: Performance evaluation of different models.

	IBLM ResNN Performance				**WRN Performance**				**CNN Performance**			
	Precision	**Recall**	**F-1**	**AUC**	**Precision**	**Recall**	**F-1**	**AUC**	**Precision**	**Recall**	**F-1**	**AUC**
No Resampling	0.8593	0.8527	0.8523	0.9115	0.8307	0.8267	0.8259	0.8597	0.8109	0.8065	0.8074	0.8837
SMOTE	0.8601	0.8599	0.8549	0.9158	0.8427	0.8366	0.8390	0.9017	0.7835	0.7783	0.7795	0.8667
Borderline SMOTE	0.8510	0.8511	0.8502	0.9104	0.8510	0.8511	0.8502	0.9104	0.7941	0.7935	0.7907	0.8757
K-Means SMOTE	0.8455	0.8400	0.8393	0.9038	0.8464	0.8473	0.8463	0.9081	0.8003	0.7933	0.7893	0.8755

While *precision* measures how often an instance that was predicted as positive for a given class is actually positive for it, *recall* measures how often a positive class instance in the dataset was predicted as a positive class instance by the classifier. In imbalanced datasets, the goal is to improve recall without hurting the precision. IBLM demonstrate the best precision values when compared to WRN and CNN. CNN lowest precision values are for SMOTE and Borderline SMOTE. While, the lowest value for WRN presicion is for No Resampling, which is expected, for IBLM and CNN the same case precision values are not the smallest ones. We assume that is related to the level of imbalance which for this dataset is not extremely strong. We can make the similar conclusion for F-1 score, where IBLM has the best performance, except for K-Means Smote resampling, with WRN being the winner. CNN F-1 scores are very low for all used resampling categories in this research. While ROC curves provide a visual method for determining the effectiveness of a classifier, the area under the ROC (AUC) is considered as some kind of standard metric for evaluating classifier performance trained with imbalance dataset distribution.

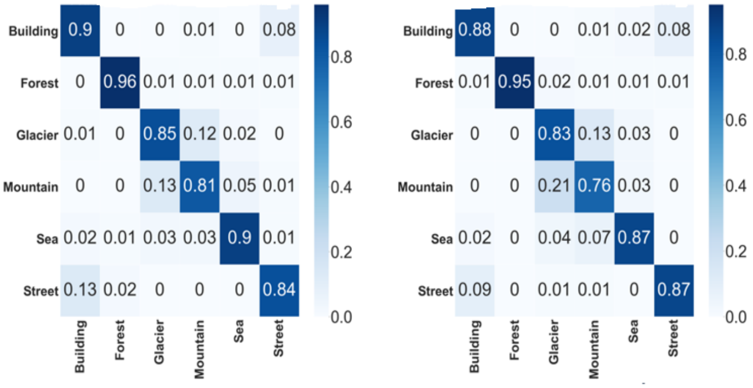
3.4. Ensemble

Table 2: Evaluation of two ensembles based on Average and Voting Individual Models.

Model	**Ensemble on Average Individual Models**				**Ensemble on Voting Individual Models**			
	Precision	**Recall**	**F-1 Score**	**AUC**	**Precision**	**Recall**	**F-1 Score**	**AUC**
All Network Types No Resampling	0.8712	0.8682	0.8685	0.9207	0.8656	0.8619	0.8621	0.9169
All Network Types SMOTE	0.8782	0.8763	0.8770	0.9256	0.8717	0.8704	0.8706	0.9220
All Network Types Borderline SMOTE	0.8657	0.8647	0.8639	0.9185	0.8550	0.8574	0.8562	0.9141
All Network Types K-Means SMOTE	0.8661	0.8677	0.8657	0.9203	0.8646	0.8664	0.8646	0.9196
All IBLM [21] Models	0.8864	0.8858	0.8856	0.9313	0.8746	0.8729	0.8721	0.9235
All WRN Models	0.8755	0.8741	0.8743	0.9242	0.8660	0.8622	0.8616	0.9171
All IBLM & WRN Models	0.8850	0.8851	0.8848	0.9308	0.8775	0.8769	0.8765	0.9259

In order to improve the classification results, we ensembled several models by using two methods: ensemble based on Average and another one on Voting [20]. Table 2 presents the performance results of each of these ensemble models with details of the included topologies and the resampling method.

Ensembling the models based on data level preprocessing strategies shown in this paper demonstrate the highest performance increase. Averaging the model's predictions is slightly better than the voting method, though both are effective. Table 2 demonstrates this difference on the SMOTE ensembles. The ensemble models achieve an approximate 8% increase in performance for all classes. Both IBLM and WRN are stronger than CNN. Ensemble combining IBLM and WRN produces better performance results than the individual models. This ensemble model achieves a much higher true positive rate than the CNN SMOTE model is able to. Fig. 7 shows the confusion matrices of Average ensemble for IBLM with SMOTE and Borderline SMOTE (left) vs. WRN (SMOTE and Borderline SMOTE – right). Test evaluation based on ROC plotting for Average ensemble of all IBLM and WRN models has a micro-average area 0.9318 and macro-average area 0.9308 vs. 0.8697 and 0.8667 for CNN with SMOTE resampling.



	Building	Forest	Glacier	Mountain	Sea	Street
Building	0.9	0	0	0.01	0	0.08
Forest	0	0.96	0.01	0.01	0.01	0.01
Glacier	0.01	0	0.85	0.12	0.02	0
Mountain	0	0	0.13	0.81	0.05	0.01
Sea	0.02	0.01	0.03	0.03	0.9	0.01
Street	0.13	0.02	0	0	0	0.84

	Building	Forest	Glacier	Mountain	Sea	Street
Building	0.88	0	0	0.01	0.02	0.08
Forest	0.01	0.95	0.02	0.01	0.01	0.01
Glacier	0	0	0.83	0.13	0.03	0
Mountain	0	0	0.21	0.76	0.03	0
Sea	0.02	0	0.04	0.07	0.87	0
Street	0.09	0	0.01	0.01	0	0.87

Fig. 7: Ensemble Average prediction of IBLM (SMOTE and Borderline SMOTE) vs. WRN (SMOTE and Borderline SMOTE).

4. Conclusions

Resampling, as an approach broadly applied in handling imbalanced datasets, requires tuning of parameters to select the proper sampling level for a given dataset. In general, this is a difficult optimization problem which depends on the size of the dataset and level of imbalance and may prove as impractical. SMOTE is a keystone for the contemporary data-level approaches for handling data imbalance, with a majority of existing oversampling approaches based directly on the idea of synthetic oversampling. We have shown that SMOTE resampling and the IBLM ResNN topology is the most effective combination for handling the imbalance when compared to WRN and CNN.

Further work can be done in fine-tuning the resampling, or other preprocessing techniques, as well as improvements to learning rate scheduling, as the latter is not addressed in this paper. Image classifiers based on algorithm modification can provide a reasonable alternative and better performance if they suggest more flexibility in implementation.

References

[1] I. Goodfellow, Y. Bengio and A. Courville, *Deep learning*, Cambridge, MA: MIT Press, 2017.

[2] J.Gu, Z,Wang, J.Kuen, L.Ma, A.Shahroudy, B.Shuai, T.Liu, X.Wang, G.Wang, J.Cai, T.Chen, "Recent advances in convolutional neural networks", *Pattern Recognition*, vol. 77, 2018, pp. 354-377. Available: 10.1016/j.patcog.2017.10.013.

[3] A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, 2017, pp. 84-90.

[4] C.Szeged, W.Liu, Y.Jia, P.Sermanet, S.Reed, D.Anguelov, D.Erhan, V.Vanhoucke, A.Rabinovich, "Going deeper with convolutions", *Proc. of the IEEE conference on computer vision and pattern recognition,* Boston, MA, 2015, pp. 1-9.

[5] P. Branco, L. Torgo, RP Ribeiro, "A Survey of Predictive Modelling on Imbalanced Distributions," *ACM Computing Surveys (CSUR),* 49 (2), 2016, pp. 1-50.

[6] H. He and Y. Ma, editors, *Imbalanced Learning: Foundations, Algorithms and Applications,* IEEE Press, 2013.

[7] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B – Cybernetics*, 42(4):1119–1130, 2012.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research,* vol. 16, pp. 321–357, 2002.

[9] H. Han, W. Wen-Yuan, M. Bing-Huan, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," *Advances in intelligent computing*, 878-887, 2005.

[10] F. Last, G. Douzas, and F. Bacao, "Oversampling for Imbalanced Learning Based on K-Means and SMOTE," *Information Sciences*, vol. 465, pp. 1-20, 2018.

[11] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "DBSMOTE: Density based synthetic minority over-sampling technique," *Applied Intelligence*, vol. 36, pp. 1–21, 2011.

[12] A. Fernández, S. García, M. Galar, R. Prati, B. Krawczyk, F. Herrera, *Learning From Imbalanced Datasets*, Springer, ISBN 978-3-319-98073-7, 2018.

[13] T. Jo and N. Japkowicz, "Class imbalances versus small disjuncts," *ACM SIGKDD Explorations Newsletter,* vol. 6, no. 1, pp. 40–49, 2004.

[14] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, "A review on ensembles for class imbalance problem: bagging, boosting and hybrid based approaches," *IEEE Trans. Syst. Man Cybern.* Part C Appl. Rev. **42**(4), 463–484, 2012.

[15] S. Hoi, R. Jin, J. Zhu, M. Lyu, "Semisupervised SVM batch mode active learning with applications to image retrieval," *ACM Trans. Inf. Syst.* **27**(3), 16:1–16:29, 2009.

[16] F. Chollet, *Deep Learning with Python*, Manning, 2018.

[17] Y. Yan, M. Chen, M. Shyu, and S. Chen, "Deep Learning for Imbalanced Multimedia Data Classification," In: *2015 IEEE International Symposium on Multimedia (ISM)*. IEEE. ISBN 978-1-5090-0379-2; p. 483–488, 2015.

[18] M. George, "Image parsing with a wide range of classes and scene-level context," In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 3622–3630, 2015.

[19] F. Tjeng, W. Cenggoro, "Deep Learning for Imbalance Data Classification using Class Expert Generative Adversarial Network," In*: 3rd International Conference on Computer Science and Computational Intelligence (ICCSCI'18)*, Indonesia, pp. 60-67, 2018.

[20] L. Rokach, "Ensemble-based classifiers,"*Artif. Intell. Rev*., vol. 33, pp. 1–39, 2010.

[21] I.Valova, C.Harris, N.Gueorguieva, T.Mai, "In-Between Layers Modular Residual Neural Network for the Classification of Images," *7th Intl Conf. of Control, Dynamic Systems, and Robotics,* 2020 (accepted submission).

[22] S. Zagoruyko, N. Komodakis, "Wide Residual Networks", *British Machine Vision Conference,* 2016.