Deep Learning for Imbalanced Multimedia Data Classification

Yilin Yan[1], Min Chen[2], Mei-Ling Shyu[1], and Shu-Ching Chen[3]
[1]Department of Electrical and Computer Engineering
University of Miami
Coral Gables, Florida, USA
[2]School of Science, Technology, Engineering & Mathematics
University of Washington Bothell
Bothell, Washington, USA
[3]School of Computing and Information Sciences
Florida International University
Miami, Florida, USA
y.yan4@umiami.edu, minchen2@uw.edu, shyu@miami.edu, chens@cs.fiu.edu

***Abstract* — Classification of imbalanced data is an important research problem as lots of real-world data sets have skewed class distributions in which the majority of data instances (examples) belong to one class and far fewer instances belong to others. While in many applications, the minority instances actually represent the concept of interest (e.g., fraud in banking operations, abnormal cell in medical data, etc.), a classifier induced from an imbalanced data set is more likely to be biased towards the majority class and show very poor classification accuracy on the minority class. Despite extensive research efforts, imbalanced data classification remains one of the most challenging problems in data mining and machine learning, especially for multimedia data. To tackle this challenge, in this paper, we propose an extended deep learning approach to achieve promising performance in classifying skewed multimedia data sets. Specifically, we investigate the integration of bootstrapping methods and a state-of-the-art deep learning approach, Convolutional Neural Networks (CNNs), with extensive empirical studies. Considering the fact that deep learning approaches such as CNNs are usually computationally expensive, we propose to feed low-level features to CNNs and prove its feasibility in achieving promising performance while saving a lot of training time. The experimental results show the effectiveness of our framework in classifying severely imbalanced data in the TRECVID data set.**

Keywords — classification; deep learning; imbalanced data; semantic indexing; convolutional neural network (CNN)

I. INTRODUCTION

Recently, class imbalance problem has attracted significant research efforts in data mining and machine learning [1]-[3]. A data set is considered imbalanced when its data instances (examples) are not close to uniformly distributed across different classes/categories, which is very common in real-world data sets. In such a data set, the class that has more data instances is defined as a majority class; while the ones with much fewer data instances are called minority classes. Since most classifiers are modeled by exploring data statistics, as a result, they may be biased towards the majority classes and hence show very poor classification accuracy on the minority classes. However, compared to the instances of the majority class, the instances of the minority classes are usually more important and interesting in a wide range of applications including rare disease in diagnosis data, fraud detection in banking operations, network intrusion detection, risk management, failure prediction of technical equipment, multimedia concept detection, etc.

To tackle this issue, many approaches have been proposed in the literature [4]-[8]. Generally speaking, they can be grouped into two categories: algorithm/model oriented approaches which aim to propose new learning mechanisms or modify existing methods to work for imbalanced data sets, and data manipulation techniques which target at changing the data distribution to make data sets less imbalanced. However, imbalanced data classification remains a challenging research problem. It is even more challenging for multimedia data due to its diverse media types and spatio-temporal characteristics.

Recent years have witnessed some important advances of new techniques in machine learning. One important breakthrough technique is known as “deep learning,” which includes a family of machine learning algorithms that attempt to model high-level abstractions in data by employing deep architectures composed of multiple non-linear transformations [9]. Many recent studies have reported encouraging results for applying deep learning approaches to a variety of applications, including speech recognition [10], object recognition [11], and natural language processing [12], among others.

However, to our best knowledge, deep learning approaches have not been applied to address the challenges in imbalanced data classification. In fact, though deep learning approaches such as convolutional neural networks (CNNs) often perform better than traditional machine learning methods in many applications, their performance can actually be worse in imbalanced data as we have observed in our empirical study (in Section IV) and presented in [13][14] on the TRECVID data (a benchmark data set with severely imbalanced data distributions). In addition, it can be computationally too expensive to apply deep learning approaches on large multimedia data sets. For example, in [15], the authors reported that they spent over a month to train the deep learning models on 1755 videos and it was almost impossible to finish their work without applying a near-duplicate finding algorithm to cut down the training set.

In this paper, we propose an extended CNN-based deep learning framework to improve multimedia data classification. In this framework, CNNs are integrated with a bootstrapping sampling algorithm which creates a set of balanced training batches, each with a few positive instances. To our best knowledge, bootstrapping has not been used to improve the performance of deep learning approaches on imbalanced data sets. In addition, our proposed bootstrapping sampling method fits the unique characteristics of CNNs so that the extended deep learning framework performs well on the data set used in the experiments. It is shown to be highly effective and efficient in classifying multimedia data with a highly imbalanced data distribution.

The rest of this paper is organized as follows. In Section 2, several imbalanced data classification approaches are discussed. Section 3 introduces the proposed framework. In section 4, experimental results and analyses are presented. Finally, the last section summarizes this paper.

II. Related Work

Imbalanced data classification approaches can be categorized to the algorithm/model oriented approaches and data manipulation techniques (in Sections II.A). These studies provide a solid theoretical foundation to extend deep learning approaches (in Section II.B) to imbalanced data classification.

A. Related work for imbalanced data classification

Algorithm/model oriented approaches mainly focused on studying and modifying the training algorithms to achieve better performance in imbalanced data classification. For instance, cost-sensitive learning methods try to maximize the loss functions associated with a data set to improve the classification performance. These learning methods are motivated by the observation that most real-world applications do not have uniform costs for misclassifications. The actual costs associated with each kind of errors are typically unknown, so these methods need to determine the cost matrix based on the data and apply it to the learning stage. A closely related idea to cost-sensitive learners is to shift the bias of a machine to favor the minority class [16]. Though some studies have shown their potential in improving classification performance on imbalanced data, they are far from extensive or systematic.

There are several types of data manipulation techniques. Among them, the sampling-based approaches including oversampling and undersampling have received significant attentions to counter the effect of imbalanced data sets [17]. Studies have tested different variants of oversampling and undersampling techniques, and presented (sometimes conflicting) viewpoints on the usefulness of oversampling versus downsampling [18] for imbalanced data sets. In oversampling, duplicate or similar positive data instances are generated by certain algorithms to make the data set balanced. Zhang et al. presented an improved oversampling approach based on the synthetic minority over-sampling technique (SMOTE) [19][20]. However, oversampling can potentially lead to overfitting. On the other hand, downsampling is to select a part of negative samples (data instances) to build a model with a similar number of positive samples. It is very efficient as it uses only a subset of the majority class. The main disadvantage is that many data instances in the majority class are ignored, which may result in the loss of information. Liu et al. proposed two algorithms to overcome this deficiency [21]. “Easy Ensemble” samples several subsets from the majority class, trains a classification model using each of them, and integrates the outputs of those models to produce the final predication results. “Balance Cascade” trains the models sequentially. In each step, the majority class data instances that are correctly classified by the current trained models are removed from the next round.

B. Recent progress in deep learning

Within the past few years, deep learning whose basic concept is originated from artificial neural network research has attracted significant research efforts in a wide range of areas including signal and information processing, machine learning, artificial intelligence, etc. due to its ability in achieving top performance for various tasks. As a result, many deep learning approaches have been studied, including Deep Belief Network (DBN), Boltzmann Machines (BM), Restricted Boltzmann Machines (RBM), Deep Boltzmann Machine (DBM), Deep Neural Networks (DNN), etc. [9]. A more detailed survey of the latest deep learning studies can be found in [22]. Among them, the convolutional neural network (CNN) [23]-[27], a discriminative deep architecture belonging to the DNN category, has achieved the state-of-the-art performance on various tasks and competitions in computer vision and image recognition. In CNNs, each module consists of a convolutional layer and a pooling layer. These modules are often stacked up with one on top of another to form a deep model. The convolutional layer shares many weights, and the pooling layer subsamples the output of the convolutional layer and reduces the data rate from the layer below.

Though CNNs have been shown with promising results for classification tasks in many applications, it remains unknown how it performs on a highly imbalanced data set. Therefore, in this paper, we investigate how effectively CNNs are for imbalanced data classification, and more importantly, how to extend it for better performance. Specifically, we propose to extend CNNs by properly integrating it with a bootstrapping sampling method that fits the unique characteristics of CNNs. Different from the negative bootstrap method in [28] that combines random sampling and adaptive selection to iteratively find relevant negatives, our proposed bootstrapping sampling method incorporates oversampling with decision fusion to enhance CNN’s performance on multimedia data classification with or without imbalanced data distributions.

III. Framework

Deep learning has become one of the most popular topics in machine learning. It is about learning multiple levels of representation and abstraction that help to make sense of the data such as images, sound, and texts. In this section, the proposed deep learning framework for imbalanced multimedia data classification is presented.

A. *Convolutional neural network*

Convolutional neural networks (CNNs) are deep learning models that are variations of multilayer perceptions designed to use minimal amounts of preprocessing [29][30] based on two ideas. The first idea is to restrict the connections between the hidden units and the input units so that each hidden unit connects to only a small subset of the input units (called feature maps in CNNs). This idea of having locally connected networks also draws the inspiration from biological discovery that neurons in the visual cortex have localized receptive fields [31]. Another idea which can be applied to reduce the computational complexity in images is that natural images have the property of being stationary. This means that the statistics of one part of the image are the same as any other part. Therefore, we can take the features learned over small patches that are randomly sampled from a large image and convolve them to obtain a different feature activation value at each location of the image. After obtaining the features using convolution, we can use them directly or use their aggregated statistics for classification. In general, the aggregated statistics are much lower in dimension (compared to using all of the extracted features) and can also improve results (less over-fitting).

Correspondingly, a convolutional network consists of several layers which can be of three types: convolutional, pooling, and fully-connected [32].

1) Convolutional layer: a convolutional layer consists of a number of feature maps. As defined in Eq. (1), the feature map at the l^{th} layer X_j^l is computed by convolving its previous layer's feature maps X_j^{l-1} through an activation function *f* with learnable kernels K_{ij}^l and additive bias b_j^l. Here, the first layer X_j^1 represents the input data, the activation function f is commonly chosen to be the logistic (sigmoid) function, and M_j represents a selection of input maps.

$$X_j^l = f\left(\sum_{i \in M_j} X_i^{l-1} * K_{ij}^l + b_j^l\right), l \geq 2; \quad (1)$$

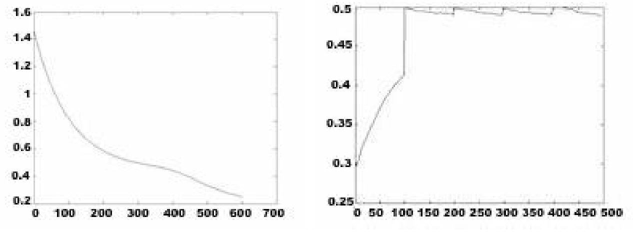
$$X_j^l = f\left(\beta_j^l pool(X_j^{l-1}) + b_j^l\right), l \geq 2. \quad (2)$$

2) Pooling layer: a pooling layer produces downsampled versions of its input feature maps as defined in Eq. (2). Here, β_j^l represents multiplicative bias, b_j^l is the additive bias, and pool(.) represents a pooling operation which generally computes the aggregated statistics of the input maps such as their mean or max values. Depending on the pooling operation applied, the layer may be called mean pooling, max pooling, etc. This layer is normally applied after each convolutional layer.

3) Fully-connected layer: after several convolutional and pooling layers, the high-level reasoning in the neural network is done via fully connected layers. A fully connected layer takes all neurons in the previous layer which may be fully connected, pooling, or convolutional and connects it to every single neuron it has.

B. *Deep learning for imbalanced data set*

Although deep learning has reached a big success in many research topics as mentioned earlier, very few approaches have been done to target it for imbalanced (multimedia) data classification. In fact, applying deep learning models directly on a skewed data set usually results in poor classification performance as shown in Figure 1, where the x-axis indicates the number of iterations and the y-axis shows the prediction error rates during the convergent process. For normal CNN processing, the prediction error rate keeps descending to a certain degree as in Figure 1(a). However, when we apply CNN on an imbalanced data set, the error rate may largely fluctuate or even increase as shown in Figure 1(b). This is because during training, most deep learning approaches, including CNNs, split the training set into several batches. However, when splitting an imbalanced data set, some of these batches may contain no positive instance but only negative ones due to the skewed distribution of the training set. As a result, such trained models perform poorly when applied to the testing set.



(a) (b)

Figure 1. Different types of total error rate convergence generated from (a) a balanced data set, and (b) an imbalanced data set.

C. *CNN with bootstrapping method*

To solve this issue, we propose to extend the CNN algorithm with a bootstrapping method. Formally, bootstrapped sampling can be defined as follows. Let *n* and *m* be the numbers of negative and positive instances, respectively, in the imbalanced training set with *n*>>*m*. In our proposed framework, each batch is created with the same size s $(s = s_n + s_p)$ and the same negative to positive ratio r $(r = s_n/s_p)$, where s_n and s_p are the numbers of negative and positive instances. Totally, *N* batches will be generated, where

$$N = \lfloor n/s_n \rfloor. \quad (3)$$

TABLE I. OVERALL PROCESS OF THE EXTENDED CNN

PSEUDO CODE OF CNN WITH BOOTSTRAPPING

```
Split the training set into a positive set pos and a negative set neg
Divide neg into N batches, each with $s_n$ negative instances
for 1:$I$
    for 1:$N$
        for 1:$s_p$
            randomly pick an instance from pos;
        end for;
        combine the pos and neg instances together;
    end for;
    Train a CNN model;
end for;
end;
```

In other words, when s_n is not exactly divisible by N, any leftover negative instances will be ignored in the training process. Since the total number of negative instances n in the training set is big and the batch size s is normally small (consequently s_n is small), the negative instances being ignored is negligible compared to the ones used in the training process. We then randomly select a positive instance from the m positive ones for s_p times and then combine them with s_n negative instances for each batch. This process will be carried out totally I times to generate batches in each training iteration. Such a random process ensures that each positive instance has an equal probability to be picked and trained with different negative instances to avoid overfitting. Table I illustrates the process. In each iteration, the bootstrapping process generates a pseudo balanced training set from the original imbalanced data. We can then use it to train the CNN model.

TABLE II. SETUP FOR CNN (DETAILED TRAINING PARAMETERS)

Layer	Layer size	Output size
Input (m*m)		
Convolution 1	k_1*n_1*n_1	k_1*$(m-n_1+1)$*$(m-n_1+1)$
Pooling 1	p_1*p_1	k_1*$(m-n_1+1)/p_1$*$(m-n_1+1)/p_1$ [let $m_2=(m-n_1+1)/p_1$]
Convolution 2	k_2*n_2*n_2	k_2*(m_2-n_2+1)*(m_2-n_2+1)
Pooling 2	p_2*p_2	k_2*$(m_2-n_2+1)/p_2$*$(m_2-n_2+1)/p_2$
Output		2

Assume the size of each input is m*m. An example four mid-layer CNN is shown in Table II, where k_L denotes the number of masks (i.e., neurons that apply on a small batch of input values) and n_L*n_L indicates the size of each mask in the L^{th} convolution layer (L = 1 or 2 in the example). The output of the L^{th} convolution layer is fed into the L^{th} pooling layer and is partitioned into a set of non-overlapping rectangles of size p_L*p_L, where the pooling operation (mean pooling in our example) is applied for down-sampling. In the literature, when applying CNNs to a multimedia data set such as images, the raw signal values of each media file (e.g., pixel intensity values of each image) are sampled and resized into m*m as the input, where m is normally a relatively big value such as 224 [33]. This makes the training process computationally demanding. In our framework, we propose to feed the low-level feature values into the CNN instead to greatly reduce the m value (e.g., 18 in our experiment) and to improve framework efficiency (see details in next section). Given the input data, the bootstrapping algorithm discussed earlier is then applied to generate N batches of balanced training data that are fed into the CNN's first layer (input layer) continuously in iterations. Following the input layers, there are two convolutional layers. Each is followed by its corresponding mean pooling layer. The first convolutional layer applies k_1 n_1*n_1 masks on the input and generates their inner product as the output. The output will be treated as the input for the first mean pooling layer. Mean pooling layers are very common in general CNNs, which summarize the outputs of the neighboring groups of masks in the same kernel map. Its output becomes the input of the second convolutional layer, followed by the mean pooling layer using the same process but with different mask sizes. The size of the final CNN output represents the number of categories to which the data are classified. In this case, we use CNNs for binary classification, and hence the size is 2.

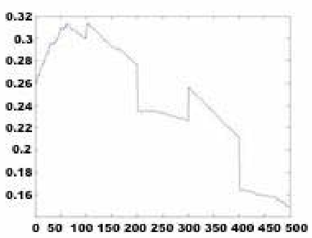


Figure 2. Total error rate convergence generated from an imbalanced data set using our bootstrapping method.

After all N batches are trained, the final convolution masks and weights will be used for classification. The testing instance will be assigned to the class with the highest output score. To illustrate the effectiveness of our bootstrapping method, it is applied to the same imbalanced data set used in Figure 1(b) and the prediction error rates are shown in Figure 2. It clearly shows the descending error rates during the convergent process.

D. Integrate CNN with low-level features

Although CNNs have been reported to perform well on several data sets, its training phase is usually time-consuming. In [15], it took a month to train 1755 videos to reach a good performance. Many groups have reported that deep learning can be computationally intensive when taking raw signal data as the input. To address this issue, we propose to use low-level features that are much smaller in sizes as the input to CNNs to reduce the computation time.

Five kinds of low-level features including Haar [34], HOG [35], HSV, YCbCr [36], and CEDD [37] were extracted and concatenated to form a feature vector. However, different from raw image pixel values that are naturally arranged in a 2-dimensional matrix layout, one-dimensional feature vector cannot be directly fed into the CNN. The principle component analysis is then applied to reduce the dimension of the feature vector to 324 for fast computation, which are further reshaped to 18*18 matrices as the input to the CNN.

TABLE III. DETAILED TRAINING PARAMETERS

Layer	Mask size	Output Size
Input (18*18)		
Convolution 1	6*3*3	6*16*16
Convolution 2	9*3*3	9*14*14
Output		2

The proposed bootstrapping method can then be applied to these matrices to generate a set of balanced inputs to the CNN. The process in the CNN is similar to the discussion in Section III.C, except that the pooling layers are removed because the low-level features are not always stationary. Table III shows the detailed training parameters used for the TRECVID data set in the experiment. A relatively small mask size was selected due to the low dimension of the feature input, in comparison to that in [33].

IV. EXPERIMENTAL RESULTS

To demonstrate the effectiveness of our proposed framework for multimedia data classification, it is tested on the TRECVID data set which is a large benchmark data set with a highly imbalanced data distribution.

A. Experimental Setup

For the experiment, the IACC.1.B data set is chosen from the TRECVID 2011 benchmark [38], whose semantic indexing (SIN) task aims to recognize the semantic concept contained within a video shot, which can be an essential technology for retrieval, categorization, and other video exploitations. Here, the concepts refer to high-level semantic objects such as a car, road, and tree. It has several challenges such as data imbalance, scalability, and semantic gap. Figure 3 shows three sample keyframes with the labeled concepts. As discussed in [13][14], traditional deep learning approaches, including CNNs, often perform poorly on the TRECVID data set due to the problem of under-fitting, huge diversity, and noisy and incomplete data annotation.

Since the data imbalance degrees of different concepts vary in the TRECVID data set, a fixed batch size may not be suitable for every testing concept. Therefore, the batch size is chosen dynamically based on the number of positive training instances in the training set. In our experiment, we set the batch size twice bigger than the number of positive training instances.

Figure 3. Sample keyframes with annotated concepts in the TRECVID data set: the concepts are bicycling, tree, and politics, respectively.

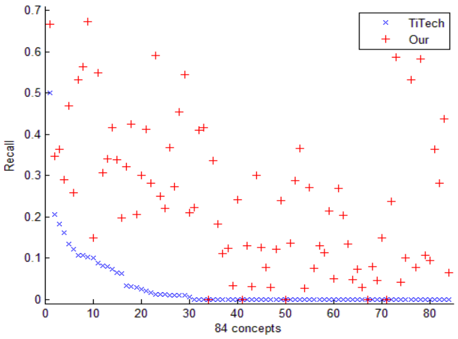


Figure 4. Recall comparsion for 84 concepts.

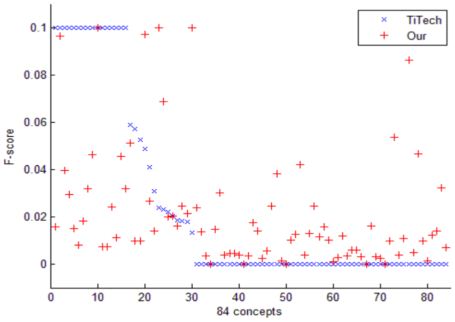


Figure 5. F-score comparsion for 84 concepts.

B. Experimental Results on the TRECVID data set

Our framework is tested on 84 concepts that are severely imbalanced with P/N ratios between 0.0001 and 0.0005. 262,911 instances are used for training and 113,046 instances are used for testing. As discussed in [39], in imbalanced data classification, the recall metric is considered more important than precision, and the F-score represents the trade-off between precision and recall. Hence, as shown in Figures 4 and 5, the recall and F-score values of our framework are compared with the scores from TiTech (Tokyo Institute of Technology) that achieved the best performance in TRECVID 2011 semantic indexing task [40][41].

As can be seen, our F-scores are higher than those of the TiTech group for 2/3rds of the 84 concepts and our recall values are much higher for almost every concept except for 4 concepts that both frameworks fail to identify any true positive instance due to the noisy and incomplete data annotations. It is also worth noting that for 50 concepts, the TiTech group can only locate zero or one true positive; while our framework reaches about 0.3 recall value on average. This clearly demonstrates the effectiveness of integrating CNNs with the bootstrapping strategy in our framework for imbalanced multimedia data classification, especially when the study in [18] showed that the performance of CNNs is far worse than all other classifiers used in their experiment on the TRECVID data set.

V. CONCULSIONS

In the paper, we propose to extend CNNs, a deep learning approach, by integrating it with a bootstrapping strategy. During the bootstrapping process, a set of pseudo balanced training batches are generated based on the properties of the data set and fed into the CNN for classification. Using the TRECVID data set, the experimental results demonstrate the effectiveness of our proposed framework in classifying multimedia data with a highly skewed data distribution. In addition, different from many existing studies in deep learning that take raw media data as the input, it is shown that our extended CNN framework can work effectively on low-level features, which greatly reduces the required training time in deep learning.

ACKNOWLEDGMENT

For Shu-Ching Chen, this research is partially supported by DHS 2010-ST-062-000039, DHS's VACCINE Center under Award Number 2009-ST-061-CI0001, NSF HRD-0833093, CNS-1126619, and NSF CNS-1461926.

REFERENCES

[1] C. Chen and M.-L. Shyu, "Integration of Semantics Information and Clustering in Binary-class Classification for Handling Imbalanced Multimedia Data," Edited by Tansel Ozyer, Keivan Kianmehr, Mehmet Tan, and Jia Zeng, *Information Reuse and Integration in Academia and Industry*, Chapter 14, Springer Verlag, 2013.

[2] C. Chen and M.-L. Shyu, "Clustering-based Binary-class Classification for Imbalanced Data Sets," *Proceedings of the 12th IEEE International Conference on Information Reuse and Integration*, pp. 384-389, Las Vegas, Nevada, USA, August 2011.

[3] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Video Semantic Concept Discovery using Multimodal-based Association Classification," *Proceedings of the IEEE International Conference on Multimedia & Expo*, pp. 859-862, Beijing, China, July 2-5, 2007.

[4] M.-L. Shyu, C. Haruechaiyasak, and S.-C. Chen, "Category Cluster Discovery from Distributed WWW Directories," *Journal of Information Sciences,* vol 155, Issues 3-4, pp. 181-197, October 2003.

[5] L. Lin, C. Chen, M.-L. Shyu, and S.-C. Chen, "Weighted Subspace Filtering and Ranking Algorithms for Video Concept Retrieval," *IEEE Multimedia*, Vol. 18, No. 3, pp. 32-43, July-September 2011.

[6] M.-L. Shyu, S.-C. Chen, M. Chen, C. Zhang, and K.Sarinnapakorn, "Image Database Retrieval Utilizing Affinity Relationships," *Proceedings of the First ACM International Workshop on Multimedia Databases*, pp. 78-85, Nov. 7, 2003, New Orleans, Louisiana, USA.

[7] D. Liu, Y. Yan, M.-L. Shyu, G. Zhao, and M. Chen, "Spatio-temporal Analysis for Human Action Detection and Recognition in Uncontrolled Environments," *International Journal of Multimedia Data Engineering and Management*, vol. 6, issue 1, pp. 1-18, 2015.

[8] K.-T. Chuang, J.-W. Hsieh, and Y. Yan, "Modeling and Recognizing Action Contexts in Persons Using Sparse Representation," *2012 International Computer Symposium*, 21, pp. 531-541, Hualien, Taiwan, December 2012.

[9] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, "Deep Learning for Content-Based Image Retrieval: A Comprehensive Study," *Proceedings of the ACM International Conference on Multimedia*, pp. 157-166, Novemember 2014.

[10] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional Neural Networks for Distant Speech Recognition," *IEEE Signal Processing Letters*, vol.21, no.9, pp.1120-1124, September 2014.

[11] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks," *IEEE Geoscience and Remote Sensing Letters*, vol.11, no.10, pp. 1797-1801, Octorber 2014.

[12] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, "Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks," *IEEE Transactions on Multimedia*, vol.16, no.8, pp. 2203-2213, December 2014.

[13] Y. Sun, T. Osawa, K. Sudo, Y. Taniguchi, H. Li, Y. Guan, and L. Liu, "TRECVid 2013 Semantic Video Concept Detection by NTT-MD-DUT," *TRECVID 2013*, November 26 – 28, 2013.

[14] C.G.M. Snoekyz, K.E.A. van de Sandeyz, D. Fontijnez, A. Habibiany, M. Jain, S. Kordumovay, Z. Liy, M. Mazloomy, S.L. Pinteay, R. Taoy, D.C. Koelmayz, and A.W.M. Smeulders, "MediaMill at TRECVID 2013: Searching Concepts, Objects, Instances and Events in Video," *TRECVID 2013*, November 26 – 28, 2013.

[15] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-Scale Video Classification with Convolutional Neural Networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 1725-1732, June 2014.

[16] C. Unsworth and G. Coghill, "Excessive Noise Injection Training of Neural Networks for Markerless Tracking in Obscured and Segmented Environments," *Neural Computation*, vol.18, no.9, pp.2122-2145, September, 2006.

[17] Y. Yan, Y. Liu, M.-L. Shyu, and M. Chen, "Utilizing Concept Correlations for Effective Imbalanced Data Classification," *Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration*, pp. 561-568, August 13-15, 2014.

[18] G. E. Batista, R. C. Prati, and M. C. Monard, "A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data," *SIGKDD Explorations*, vol. 6, issue 1, pp. 20-29, June 2004.

[19] L. Zhang and W. Wang, "A Re-sampling Method for Class Imbalance Learning with Credit Data," *Proceedings of the 2011 International Conference on Information Technology, Computer Engineering and Management Sciences*, pp. 393-397, September 2011.

[20] N. V. Chawla and K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Overbootstrapping Technique," *Journal of Artijcial Intelligence Research*, 16, pp. 321-357, 2002.

[21] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory Undersampling for Class-Imbalance Learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(2), pp.539-550, April 2009.

[22] L. Deng, "A Tutorial Survey of Architectures, Algorithms, and Applications for Deep Learning," *APSIPA Transactions on Signal and Information Processing*, vol.3, no.2, 2014.

[23] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional Neural Networks for Distant Speech Recognition," *IEEE Signal Processing Letters*, vol.21, no.9, pp.1120-1124, September 2014.

[24] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks," *IEEE Geoscience and Remote Sensing Letters*, vol.11, no.10, pp. 1797-1801, Octorber 2014.

[25] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, "Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks," *IEEE Transactions on Multimedia*, vol.16, no.8, pp. 2203-2213, December 2014.

[26] S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.35, no.1, pp.221-231, January 2013.

[27] J. Jin, K. Fu and C. Zhang, "Traffic Sign Recognition With Hinge Loss Trained Convolutional Neural Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol.15, no.5, pp.1991-2000, Oct. 2014.

[28] X. Li, C. G. M. Snoek, M. Worring, D. Koelma and A. W. M. Smeulders, "Bootstrapping Visual Categorization With Relevant Negatives," *IEEE Transactions on Multimedia*, volume 15, issue4, pp. 933-945, 2013.

[29] T. Hastie, "Neural Networks," Edited by P. Armitage and T. Colton, Encyclopedia of Biostatistics, John Wiley & Sons, 2005, ISBN 9780470011812.

[30] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol.86, no.11, pp.2278-2324, November 1998.

[31] E. R. Kandel, "An Introduction to the Work of David Hubel and Torsten Wiesel," *The Journal of Physiology* 587 (Pt 12), pp. 2733–2741, April 2009.

[32] J. Bouvrie, "Notes on Convolutional Neural Networks.," *Technical report*, 2006.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *Proceeding of the European Conference on Computer Vision*, pp. 346-361, Zurich, Switzerland, September 6-12, 2014.

[34] D. Verma and V. Maru. "An Efficient Approach for Color Image Retrieval using Haar Wavelet," *Proceeding of the IEEE International Conference on In Methods and Models in Computer Science*, pp. 1–5, 2009.

[35] Y. Yan, J.-W. Hsieh, H.-F. Chiang, S.-C. Cheng, and D.-Y. Chen, "PLSA-Based Sparse Representation for Object Classification," *Proceedings of the 2014 22nd International Conference on Pattern Recognition*, pp. 1295-1300, August 24-28, 2014.

[36] S. Sural, G. Qian, and S. Pramanik, "Segmentation and Histogram Generation using the HSV Color Space for Image Retrieval," *Proceeding of the International Conference on Image Processing (ICIP)*, pp. 589-592, 2002.

[37] S. A. Chatzichristofis and Y. S. Boutalis, "CEDD: Color and Edge Directivity Descriptor: A Compact Descriptor for Image Indexing and Retrieval," *In Proceedings of the 6th international conference on Computer vision systems, ICVS'08*, pp. 312–322, Berlin, Heidelberg, 2008.

[38] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation Campaigns and TRECVid," *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pp. 321-330, 2006.

[39] G. Qiong, Z. Li, and C. Zhihua, "Evaluation Measures of the Classification Performance of Imbalanced Data Sets," *Proceedings of the 4th International Symposium, ISICA 2009,* pp. 461-471, October, 2009.

[40] N. Inoue, T. Wada, Y. Kamishima, K. Shinoda, and S. Sato, "TokyoTech+Canon at TRECVID 2011," *Proceedings of the TRECVID Workshop 2011*, December 5, 2011.

[41] N. Inoue and K. Shinoda, "A Fast and Accurate Video Semantic-Indexing System Using Fast MAP Adaptation and GMM Supervectors," *IEEE Transactions on Multimedia*, vol. 14, no. 4-2, pp. 1196-1205, 2012.