TRANSFERRING CNNS TO MULTI-INSTANCE MULTI-LABEL CLASSIFICATION ON SMALL DATASETS

Mingzhi Dong[1], *Kunkun Pang*[2], *Yang Wu*[3*], *Jing-Hao Xue*[1], *Timothy Hospedales*[2], *Tsukasa Ogasawara*[3]

[1]University College London [2] University of Edinburgh [3]Nara Institute of Science and Technology

ABSTRACT

Image tagging is a well known challenge in image processing. It is typically addressed through multi-instance multi-label (MIML) classification methodologies. Convolutional Neural Networks (CNNs) possess great potential to perform well on MIML tasks, since multi-level convolution and max pooling coincide with the multi-instance setting and the sharing of hidden representation may benefit multi-label modeling. However, CNNs usually require a large amount of carefully labeled data for training, which is hard to obtain in many real applications. In this paper, we propose a new approach for transferring pre-trained deep networks such as VGG16 on Imagenet to small MIML tasks. We extract features from each group of the network layers and apply multiple binary classifiers to them for multi-label prediction. Moreover, we adopt an L_1-norm regularized Logistic Regression (L_1LR) to find the most effective features for learning the multi-label classifiers. The experiment results on two most-widely used and relatively small benchmark MIML image datasets demonstrate that the proposed approach can substantially outperform the state-of-the-art algorithms, in terms of all popular performance metrics.

Index Terms— CNN, Multi-instance, Multi-label, Small dataset, Transfer Learning

1. INTRODUCTION AND RELATED WORK

Abstracting meaningful visual information from images into semantics is one of the popular research areas in image processing, namely, feature extraction. A fundamental but unsolved problem is how to automatically generate text interpretation or description of images, an ultimate goal of image understanding. Publicly accessible images are usually posted for transmitting information, so each of them is likely worth many (if not a thousand) words. If some of these words are treated as labels, then the prediction of them becomes a multi-label learning problem. Meanwhile, each image can be viewed as a bag of local regions. If labels are assigned to a whole image but not its specific regions, i.e. an image (bag) label is positive if any of its regions (instances) has this positive label, the prediction of each image (bag) label is also naturally a multi-instance learning problem. Therefore, the so-called multi-instance multi-label (MIML) problem [1] is in fact very common in real life and of great importance for bridging images and text in multimedia researches. Figure 1 illustrates a general setting for automatic image tagging (annotation) in the MIML perspective[1]. Recently, Convolutional Neural Networks (CNNs) have been successfully applied to numerous computer vision tasks, such as face recognition [2], image categorization [3], scene recognition [4], and semantic segmentation [5]. These tasks are still essentially a multi-class classification problem, where only a single label is assigned to a concerned input, not yet a MIML problem.

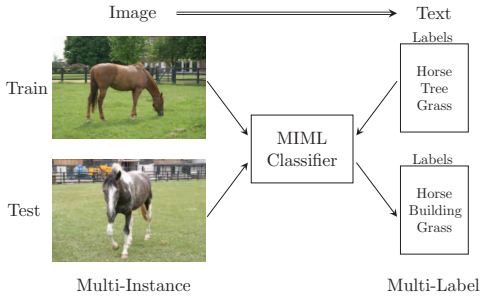

Fig. 1: Multi-instance multi-label learning for image tagging.

However, here we would like to promote CNNs further that CNNs are in fact suitable for solving MIML problems. Firstly, traditional algorithms solve multi-instance problems by extracting a bag of instances from each image and then applying multi-instance classifiers. The convolutional layers of CNNs slide through an image (the bag) and create the candidate instances, and the max-pooling layers select the most representative instances inside the bag. Secondly, traditional algorithms solve multi-label problems by explicitly or implicitly building the relationship between the labels. In CNNs, through utilizing a deep hierarchical structure, different levels of representation of all the labels have already been embedded in the network, and the relationship of the labels can be explored via checking the sharing of the neurons, for exam-

*Corresponding author. This work was supported by JSPS KAKENHI Grant Number 15K16024 and the National Natural Science Foundation of China Under Grant No.61373077.

[1]In some related work multi-instance learning also concerns instance-level labeling, but in this paper we focus on bag-level labeling.

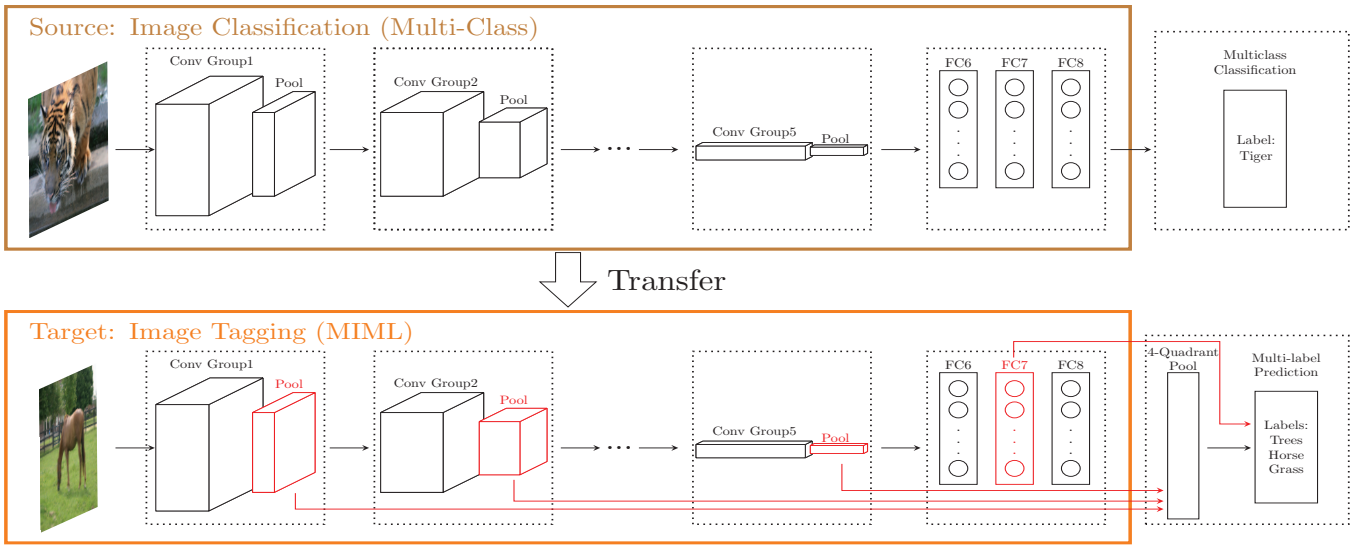

Fig. 2: The framework of the learning algorithm. Source Task: Multi-class Classification (Image Classification), outputting one class label; Target Task: MIML Recognition (Image Tagging), outputting multiple labels.

ple. Hence a simple way to adapt CNNs for solving MIML problems would be to change the original multi-class classifier in the Softmax Layer into a multi-label classifier, such as the multiple binary-class logistic regression (LR).

Unfortunately, unlike multi-class classification datasets such as Imagenet [3], most existing MIML datasets only have a relatively small amount of training data, as the annotation costs for them are generally much higher. Hence, it will be hard to directly train an effective CNN model for such cases.

In this paper, we propose a way to transfer existing deep CNN models to small MIML tasks. Based on the VGG 16 layers network (VGG16) [6] pre-trained on Imagenet, we extract features from each group of its layers, which enable a depiction of the multi-level relationship between the labels. Then an L_1-norm regularized Logistic Regression (L_1LR) is adopted to learn one classifier for each label. The aim of utilizing the sparsity regularization is to encourage the classifiers to select only much smaller subsets of "effective" features for specific labels. We shall use experiments on two most-widely used small benchmark MIML image datasets to show that the proposed approach, as well as its simpler version without L_1 regularization, can substantially outperform the state-of-the-art algorithms, in terms of all of the popular metrics for evaluation of MIML classification, namely the Hamming loss, one error, coverage, ranking loss and average precision [1].

2. TRANSFER LEARNING FOR MIML

To use a representative and concrete exemplar model to illustrate the transferring of CNNs, in this paper we choose the VGG16 net [6] trained on the multi-class classification task of Imagenet. VGG16 is a widely-used simple yet powerful CNN structure, enjoying good performance on Imagenet and many successful transferring records. Imagenet has been proved to be a great source for transferring as it has a wide coverage (totally 1000) of common object categories and contains great variations of them in its 1.3 million training images.

VGG16 has 13 convolutional layers and 3 Fully-Connected (FC) layers. The 13 convolutional layers form 5 groups (2-2-3-3-3) and at the end of each group one max-pooling layer is utilized. During the training process, VGG16 learns filters/kernels, which would slide through each image at multiple scales and generate nearly all possible instances (patches) inside the bag (image). Max pooling, which would add position robustness, selects the most representative instances inside the bag in fact. Therefore, the features learned via CNNs are potentially suitable for multi-instance tasks.

Meanwhile, during the training process with Imagenet, a larger number of labels have been simultaneously employed to learn a representation network. The hierarchical structure of the network builds different levels of representation of all the labels. Later layer filters, which are semantic detectors [7, 8], are in fact closely related to the semantic description space of the labels [5, 9]. Earlier layer filters would be more about the low level descriptions of the labels [10]. With the pre-trained network, different levels of semantic relationship have already been captured by the network via the sharing of neurons at different layers. Therefore, the features learned via CNNs are also suitable for multi-label tasks on small datasets. One widely adopted approach to transfer learning is fine-tuning. However, an effective fine-tuning usually also requires a large number of training instances [11] or pixel-level strongly supervised information [5]. Hence, for small MIML image tagging tasks, we introduce the following transfer learning method.

According to the previous papers such as [12, 13, 14, 15],

Table 1: Performance comparison (mean ± std). The symbol ↑ (↓) indicates that the larger (smaller) the value, the better the performance.

	VGG+L_1LR	VGG+LR	MIMLfast	DBA	KISAR	MIMLkNN	MIMLSVM	RankLSIM
MSRC v2								
a.p. ↑	.933 ± .010	.851 ± .016	.688 ± .017	.326 ± .016	.666 ± .018	.591 ± .018	.685 ± .018	.687 ± .013
co. ↓	.102 ± .006	.140 ± .012	.238 ± .014	.837 ± .018	.254 ± .015	.312 ± .020	.256 ± .018	.239 ± .013
h.l. ↓	.033 ± .003	.051 ± .003	.100 ± .007	.140 ± .006	.086 ± .004	.131 ± .007	.084 ± .003	.110 ± .004
o.e. ↓	.060 ± .017	.149 ± .024	.295 ± .025	.415 ± .026	.341 ± .031	.440 ± .031	.320 ± .029	.302 ± .208
r.l. ↓	.018 ± .003	.045 ± .007	.108 ± .009	.675 ± .017	.131 ± .010	.165 ± .013	.125 ± .011	.107 ± .007
Scene								
a.p. ↑	.948 ± .006	.926 ± .004	.770 ± .015	.600 ± .013	.772 ± .012	.757 ± .011	.750 ± .012	.738 ± .011
co. ↓	.082 ± .006	.096 ± .004	.207 ± .012	.334 ± .011	.204 ± .008	.222 ± .009	.225 ± .010	.237 ± .010
h.l. ↓	.070 ± .004	.090 ± .004	.188 ± .009	.269 ± .009	.194 ± .005	.196 ± .007	.200 ± .008	.204 ± .007
o.e. ↓	.082 ± .010	.114 ± .007	.351 ± .023	.386 ± .025	.351 ± .020	.370 ± .018	.380 ± .021	.392 ± .019
r.l. ↓	.038 ± .005	.055 ± .004	.189 ± .014	.348 ± .012	.185 ± .010	.207 ± .011	.212 ± .011	.222 ± .010

features from all the layers are useful for the final classification. More precisely, they extract features from all layers and then conduct 4-quadrant max-pooling for each filter of the convolution layers. Similarly, we use 4-quadrant max-pooling to extract features from the pooling layer after each group of convolution layers, as illustrated in Figure 2. The second, instead of the last, FC layer is adopted, because the last layer may be too specific for the final tasks and the second one may be better for transferring. The features from pooling layers can represent instance-level information as the convolutions can be regarded as extracting filter-specific features for instances and the pooling is about selecting the most representative instances. Differently, the FC layer mixes all instances and generates a bag-level representation so it may cover certain relationships between different instances. Both the instance-level and bag-level representations are considered to be important for multi-instance (MI) learning under both traditional MI assumption or its generalized assumptions [16]. The final features have a total dimension of 9984, with 64×4, 128×4, 256×4, 512×4, 512×4, and 4096 for the corresponding components, respectively.

A small MIML dataset usually contains few labels, which may be related to only a small subset of the Imagenet classes. Therefore, many of the pre-trained CNN filters may not be relevant to the MIML problem. So intuitively it is natural to conduct L_1 regularization to obtain the effective subsets of features for specific MIML tasks. Meanwhile, to carry out multi-label classification, we replace the original classifier of multi-class LR (in the Soft-max Layer, which would encourage the classifier to output a single class label) with multiple binary LR classifiers (each corresponds to a label). Therefore, the optimization problem to learn the j-th classifier can be expressed as

$$\min_{\boldsymbol{w}_j} \frac{1}{m} \sum_{i=1}^{m} \log(1 + \exp(-y_{ij}(\boldsymbol{x}_i^T \boldsymbol{w}_j + b))) + \lambda |\boldsymbol{w}_j|_1 \quad (1)$$

where $\boldsymbol{x}_i \in R^F$ represents the extracted features of the ith instance; $\boldsymbol{w}_j \in R^F$ denotes the LR parameters for label j; b is the intercept; $y_{ij} \in \{-1, 1\}$ indicates the multi-label supervised information, where $y_{ij} = 1$ (or -1) if the ith instance has label j (or not); m denotes the number of training instances; and λ is the shrinkage parameter. This simple classification model implicitly takes multi-instance learning into account, given that the CNN features extracted and employed have already embedded MIML information, as explained.

3. EXPERIMENTS

3.1. Setting

We use VGG+L_1LR (VGG+LR) to represent the transferring algorithm adopting the features extracted from VGG16 and then using L_1LR (LR) to do multi-label classification. VGG+L_1LR and VGG+LR are compared with other six state-of-the-art MIML methods, namely MIMLfast [17], DBA [18], KISAR [19], MIMLkNN [20], MIMLSVM [21] and RankLSIM [22].

The experiments are conducted on two popular and relatively small benchmark datasets: MSRC v2 [23], a subset of the Microsoft Research Cambridge (MSRC) image dataset with 591 images and 23 labels (on average 2.5 labels per image), and Scene [21], a natural scene dataset with 2000 images and 5 different labels (on average 1.2 labels per image).

The VGG features are extracted via Caffe [24]; LR and L_1LR are implemented via Liblinear [25]. The shrinkage parameter λ of L_1LR is fixed as default of 1. The performance of those state-of-the-art algorithms are based on [17], which adopts cross validation to select suitable values for the parameters of the algorithms. For each dataset, $2/3$ of the instances are randomly selected to train the classifiers and the remaining $1/3$ instances are taken as the test instances. We repeat the random data partition for 30 times, and report the average results over the 30 repetition.

The performance evaluation of multi-label algorithms is more complex than that of multi-class ones. We choose five

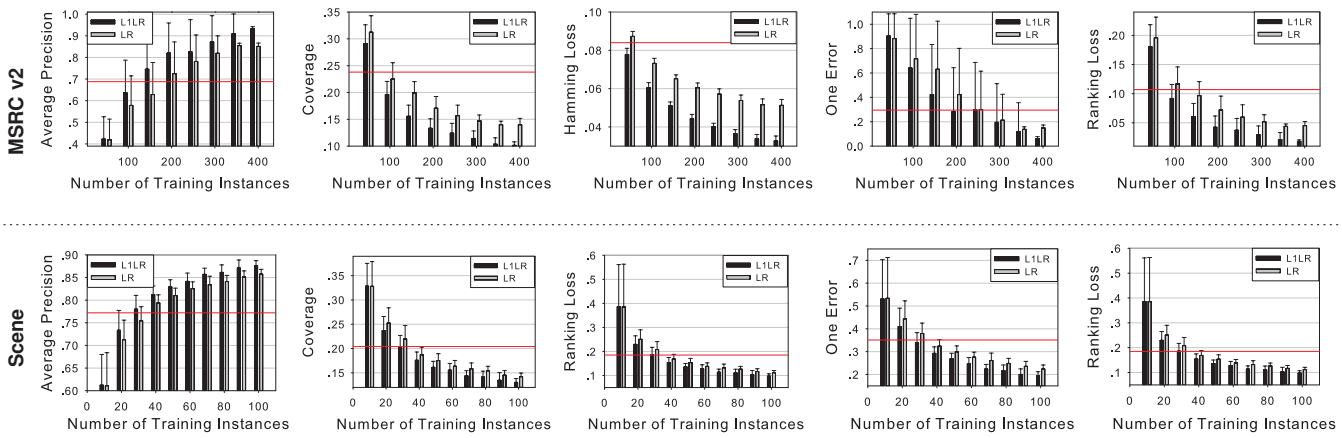

Fig. 3: The performance with a relatively small number of training instances. The red line indicates the best performance of the 6 state-of-the-art algorithms with 2/3 training instances (394 training instances on MSRC dataset on the upper 5 panels, and 1333 training instances on Scene dataset on the lower 5 panels). The performance is based on 30 random repetition and the bar indicates the standard deviation.

frequently adopted metrics [1] for evaluation as follows.

- Average Precision (a.p.): the average fraction of labels ranked above a particular label $y \in Y$ which are actually in Y;
- Coverage (co.): how far we need, on average, to go down the list of labels in order to cover all the proper labels of the instance[2];
- Hamming Loss (h.l.): the percentage of instance-label pairs which are misclassified;
- One Error (o.e.): the average number of the top-ranked label predicted but not a label of the instance;
- Ranking Loss (r.l.): the average fraction of label pairs that are reversely ordered for the instance.

3.2. Results

Firstly, as illustrated in Table 1, it is clearly that both VGG+L_1LR and VGG+LR outperform the 6 state-of-the-art algorithms remarkably, which proves the proposed VGG-based transfer learning framework is suitable for MIML problems and could perform well even with a small number of training instances (394 on the dataset of MSRC and 1333 on the dataset of Scene). In terms of average precision, which is the higher the better, the VGG-based algorithms perform at least 0.16 and 0.15 better on the two datasets respectively. In terms of the other four metrics, namely coverage, hamming loss, one error and ranking loss, the VGG-based algorithms are at least 0.098, 0.033, 0.146, 0.062 smaller than the state-of-the-art algorithms on the dataset of MSRC and 0.111, 0.098, 0.237, 0.130 smaller on the dataset of Scene. The experiment results on these two datasets indicate that the proposed algorithms, which are based on transferring VGG, enjoy at least 21.8%, 41.1%, 33%, 49.4% and 57.4% performance improvement on the five metrics, respectively.

Secondly, after comparing VGG+L_1LR against VGG+LR, the sparsity regularization also has shown noticeable positive effect. On the dataset of MSRC, we can see $0.082(9.6\%)$, $0.038(27.1\%)$, $0.018(35.29\%)$, $0.089(59.7\%)$ and $0.027(60\%)$ improvement on the five metrics, respectively. Similarly, $0.022(2.4\%)$, $0.014(14.6\%)$, $0.020(22.2\%)$, $0.032(28.1\%)$ and $0.027(30.9\%)$ improvement is observed on the dataset of Scene. The results verify the effectiveness of sparsity regularization.

Finally, the performance of the algorithms with even smaller numbers of training instances is illustrated via Figure 3. We can see that adding L_1 regularization always produces better performance when the number of training instances is relatively small. On the dataset of MSRC, when comparing against the state-of-the-art algorithms, we can observe that, with only 200 training instances, VGG+L_1LR can perform better than the state-of-the-art algorithms with 394 training instances, in terms of all five metrics. Even more striking, on the dataset of Scene, it is clearly that with only 30 training instances, L_1LR could get better performance than the state-of-the-art algorithms with 1333 training instances. These results verify that the proposal also works with a relatively small number of training instances.

4. CONCLUSION

In this paper, for multi-instance multi-label problems with small datasets, we propose a CNN-based transfer learning framework with sparsity regularization. The proposal has achieved substantial improvement in classification performance when compared with six state-of-the-art algorithms.

[2]In this paper, the reported coverage is normalized by the number of labels such that all criteria are in the interval of [0, 1].

5. REFERENCES

[1] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li, "Multi-instance multi-label learning," *Artificial Intelligence*, vol. 176, no. 1, pp. 2291–2320, 2012.

[2] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[4] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in neural information processing systems*, 2014, pp. 487–495.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[6] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[7] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba, "Object detectors emerge in deep scene CNNs," *arXiv preprint arXiv:1412.6856*, 2014.

[8] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic, "Is object localization for free?–Weakly-supervised learning with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 685–694.

[9] Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1796–1804.

[10] Pierre-André Savalle, Stavros Tsogkas, George Papandreou, and Iasonas Kokkinos, "Deformable part models with CNN features," in *European Conference on Computer Vision, Parts and Attributes Workshop*, 2014.

[11] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3973–3981.

[12] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox, "Discriminative unsupervised feature learning with convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2014, pp. 766–774.

[13] Will Zou, Shenghuo Zhu, Kai Yu, and Andrew Y Ng, "Deep learning of invariant features via simulated fixations in video," in *Advances in neural information processing systems*, 2012, pp. 3212–3220.

[14] Ka Y Hui, "Direct modeling of complex invariances for visual object features," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, pp. 352–360.

[15] Adam Coates and Andrew Y Ng, "Selecting receptive fields in deep networks," in *Advances in Neural Information Processing Systems*, 2011, pp. 2528–2536.

[16] James R. Foulds and Eibe Frank, "A review of multi-instance learning assumptions.," *Knowledge Eng. Review*, vol. 25, no. 1, pp. 1–25, 2010.

[17] Sheng-Jun Huang, Wei Gao, and Zhi-Hua Zhou, "Fast multi-instance multi-label learning," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, 2014, pp. 1868–1874.

[18] Shuang-Hong Yang, Hongyuan Zha, and Bao-Gang Hu, "Dirichlet-Bernoulli alignment: A generative model for multi-class multi-label multi-instance corpora," in *Advances in neural information processing systems*, 2009, pp. 2143–2150.

[19] Yu-Feng Li, Ju-Hua Hu, Yuan Jiang, and Zhi-Hua Zhou, "Towards discovering what patterns trigger what labels," in *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, 2012.

[20] Min-Ling Zhang, "A k-nearest neighbor based multi-instance multi-label learning algorithm," in *Tools with Artificial Intelligence (ICTAI), 2010 22nd IEEE International Conference on*. IEEE, 2010, vol. 2, pp. 207–212.

[21] Zhi-Hua Zhou and Min-Ling Zhang, "Multi-instance multi-label learning with application to scene classification," in *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, 2006, pp. 1609–1616.

[22] Forrest Briggs, Xiaoli Z Fern, and Raviv Raich, "Rank-loss support instance machines for miml instance annotation," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 534–542.

[23] John Winn, Antonio Criminisi, and Thomas Minka, "Object categorization by learned universal visual dictionary," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. IEEE, 2005, vol. 2, pp. 1800–1807.

[24] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[25] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, "Liblinear: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.