FAST TAGGING OF NATURAL SOUNDS USING MARGINAL CO-REGULARIZATION

Qiang Huang, Yong Xu, Philip J. B. Jackson, Wenwu Wang, Mark D. Plumbley

Centre for Vision, Speech and Signal Processing,
University of Surrey, Guildford, GU2 7XH, UK
{q.huang, yong.xu, p.jackson, w.wang, m.plumbley}@surrey.ac.uk

ABSTRACT

Automatic and fast tagging of natural sounds in audio collections is a very challenging task due to wide acoustic variations, the large number of possible tags, the incomplete and ambiguous tags provided by different labellers. To handle these problems, we use a co-regularization approach to learn a pair of classifiers on sound and text. The first classifier maps low-level audio features to a true tag list. The second classifier maps actively corrupted tags to the true tags, reducing incorrect mappings caused by low-level acoustic variations in the first classifier, and to augment the tags with additional relevant tags. Training the classifiers is implemented using marginal co-regularization, pair of which draws the two classifiers into agreement by a joint optimization. We evaluate this approach on two sound datasets, Freefield1010 and Task4 of DCASE2016. The results obtained show that marginal co-regularization outperforms the baseline GMM in both efficiency and effectiveness.

Index Terms— natural sound, annotation, co-regularization

1. INTRODUCTION

With the use of mobile devices in recent years, about 2.5 billion gigabytes of data are generated and uploaded to the internet everyday [1]. The need to analyze the data has been increasing as it is a very valuable resource for audio and visual analysis and processing. Currently, research and applications in audio rely heavily on annotations of audio data, so fast and automatic annotation of large sound collections is desirable. However, the wide acoustic variations, large size of the set of tag, imprecise or incomplete tags provided by users, and the need to fast tagging, make this task very challenging.

In this work, we aim to explore fast and automatic tagging of natural sounds, where we have features from two different modalities, sound and text (tags). Inspired by some studies in image annotation [2, 3, 4], we will use a co-regularization framework [5, 2]. Its main idea is to train the target model with multiple distinct feature sets rather than a single feature set. It works by making the hypotheses learned from different feature sets agree with each other on the same target [6]. Minimization of a co-regularization term has been shown to be an effective way to reduce disagreement of different classifiers [5, 7]. Figure 1 shows our co-regularization framework consisting of the training and test step. In the training step, a pair of classifiers are trained. The first classifier maps the given low-level audio features (**X**) to a true tag list (**Y**). The second classifier uses actively reduced tags ($\widehat{\mathbf{Y}}$) as input and maps them to the true tags. This aims to reduce incorrect mappings caused by low-level acoustic variations in the first classifier. In the test step, the learned model can output tags given the input sound.

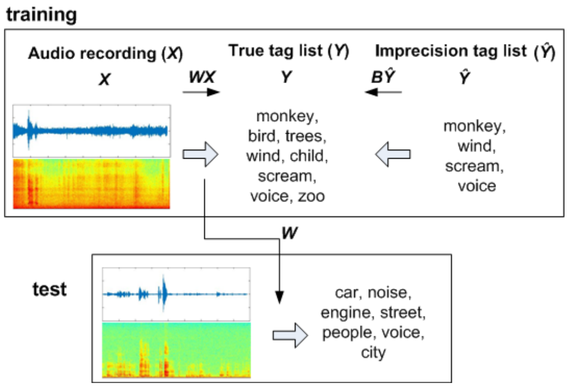
training
Audio recording (*X*)
True tag list (*Y*)
Imprecision tag list (*Ŷ*)
X
WX
Y
BŶ
Ŷ
monkey, bird, trees, wind, child, scream, voice, zoo
monkey, wind, scream, voice
W
test
car, noise, engine, street, people, voice, city

Fig. 1. Framework of sound tagging using co-regularization.

Although audio analysis has been widely studied in scene classification [8, 9, 10], audio segmentation [11, 12, 13], and audio retrieval [14, 15, 16], to our knowledge, automatic audio tagging has not been much explored. Bertin-Mahieux et al. [17] treated audio tag prediction as a set of binary classification problems and applied the Adaboost algorithm to the task. In [18], Panagakis et al. addressed the problem of automatic music tagging as multi-label multi-class classification problem, and employed a multilinear subspace learning algorithm and sparse representations. Recently, Task4 of DCASE2016 [19] involved automatic audio tagging on a small sound dataset recorded in a specific domestic environment. However, in these previous studies there are limitations on audio tagging in complex conditions. First, they mainly

The work is funded by EPSRC grant EP/N014111/1.

focused on specific audio signals, such as music, or sound recorded in a specific environment. This means the acoustic variations might be narrower within sound classes in comparison with large natural sound collections. Second, the number of sound classes in these datasets is not large, and thus makes the size of tag set relatively small as well. Third, the sound datasets used in these work were well annotated without any imprecise, incomplete and redundant tagging, which is not typical of other online data.

In comparison with the ongoing research in automatic sound tagging, image tagging has been well studied in two aspects, namely tag assignment and tag refinement. Image tag assignment strives to assign a number of tags related to the image content to unlabeled images [2, 20, 21, 22]. Image tag refinement aims to remove irrelevant tags from the initial tag list and enrich it with novel, yet relevant tags [23, 24, 25]. Inspired by the technologies used in image tagging, we will use the co-regularization framework, as shown in figure 1, to fast and automatically tag large natural sounds. The details will be given in the next section.

2. THEORETICAL FRAMEWORK

Given the training audio recordings $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$, our goal is to learn a model to map the audio features to tags $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\}$, where each vector $\mathbf{x}_i$ represents the features extracted from the i^{th} audio recording and $\mathbf{y}_i$ denotes the tags corresponding to $\mathbf{x}_i$.

2.1. Coregularized Learning

To automatically annotate audio data, we use a co-regularized learning algorithm [26, 2]: 1) training a mapping, $\mathbf{x}_i \rightarrow \mathbf{W}\mathbf{x}_i$, that predicts the tags given the audio features 2) training a mapping $\widehat{\mathbf{y}}_i \rightarrow \mathbf{B}\widehat{\mathbf{y}}_i$ to enrich the existing incomplete tag, and thus to reduce the possible incorrect mappings caused by low-level acoustic variations in the first step. The two classifiers are jointly optimized by minimizing

$$\frac{1}{n}\sum_{i}^{n} \parallel \mathbf{B}\widehat{\mathbf{y}}_i - \mathbf{W}\mathbf{x}_i \parallel^2 \tag{1}$$

where $\mathbf{B}\widehat{\mathbf{y}}_i$ is the adapted tag set for the i^{th} training audio recording, and each row of $\mathbf{W}$ contains the weights of a linear classifier that tries to predict the corresponding tags given audio features. The basic working flow is shown in figure 1.

The optimization of $\mathbf{W}$ and $\mathbf{B}$ are run simultaneously. For $\mathbf{W}$, we use the ℓ_2 regularizer. For $\mathbf{B}$, we use the marginalized blank-out algorithm [27]. The joint loss function can therefore be written as

$$\ell(\mathbf{B}, \mathbf{W}; \mathbf{x}, \mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n} \parallel \mathbf{B}\widehat{\mathbf{y}}_i - \mathbf{W}x_i \parallel^2 + \lambda \parallel \mathbf{W} \parallel_2^2 + \gamma f(\mathbf{B}) \tag{2}$$

The first term represents a joint optimization to enforce the prediction of true tags by adapting the existing labels and making them agree with the tags predicted by the given audio recordings. A regularizer on $\mathbf{W}$ is included to reduce complexity and avoid over-fitting. The last term ensures that the adaptation mapping $\mathbf{B}$ reliably predicts tags if they were to be removed from the training label set [2].

2.2. Marginalized blank-out regularization

From denoising point of view, the incomplete tags $\widehat{\mathbf{Y}}$ can be viewed as a "corrupted" version of the true tags $\mathbf{Y}$, and training $\mathbf{B}$ is for denoising. If this denoising mechanism works, then the use of $\mathbf{B}$ would recover the likely original tags. During training in our experiments, we assume the observed tag is a true tag ($\mathbf{Y}$), and a corrupted version $\widehat{\mathbf{Y}}$ is created by randomly removing each entry in $\mathbf{Y}$ with some probability $p \geq 0$; accordingly, for each tag vector $\mathbf{Y}$ and dimensions t, $p(\widehat{\mathbf{y}}_t = 0) = p$ and $p(\widehat{\mathbf{y}}_t = \mathbf{y}_t) = 1 - p$ (p and λ are set 0.1, and γ is set 0.01 in our experiments). $\mathbf{B}$ can then be optimized by [2]:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \frac{1}{n}\sum_{i=1}^{n} \parallel \mathbf{y}_i - \mathbf{B}\widehat{\mathbf{y}}_i \parallel^2 \tag{3}$$

Here, each row of $\mathbf{B}$ is an ordinary least squares regressor that predicts the presence of a tag given all existing tags in $\widehat{\mathbf{y}}$. The expected reconstruction error $f(\mathbf{B})$ under the corrupting distribution can be expressed as [2]:

$$f(\mathbf{B}) = \frac{1}{n}\sum_{i}^{n} E[\parallel \mathbf{y}_i - \mathbf{B}\widehat{\mathbf{y}}_i \parallel^2] \tag{4}$$

After defining $\mathbf{P} = \sum_{i=1}^{n} \mathbf{y}_i E[\widehat{\mathbf{y}}_i]^\top$ and $\mathbf{Q} = \sum_{i=1}^{n} E[\widehat{\mathbf{y}}_i \widehat{\mathbf{y}}_i^\top]$, we rewrite the loss in (4) as [2]:

$$f(\mathbf{B}) = \frac{1}{n} trace(\mathbf{B}\mathbf{Q}\mathbf{B}^\top - 2\mathbf{P}\mathbf{B}^\top + \mathbf{Y}\mathbf{Y}^\top) \tag{5}$$

For the uniform "blank-out" noise introduced above, we have the expected value of the corruptions $E[\widehat{\mathbf{y}}] = (1-p)\mathbf{y}$ and the variance matrix $V[\widehat{\mathbf{y}}] = p(1-p)\delta(\mathbf{y}\mathbf{y}^\top)$. Here δ means that the variance matrix has non-zero entries only on the diagonal. The computation of $\mathbf{P}$ and $\mathbf{Q}$ is then done using the following equations, respectively [2].

$$\begin{aligned}\mathbf{P} &= (1-p)\mathbf{Y}\mathbf{Y}^\top \\ \mathbf{Q} &= (1-p)^2\mathbf{Y}\mathbf{Y}^\top + p(1-p)\delta(\mathbf{Y}\mathbf{Y}^\top)\end{aligned}$$

$\mathbf{W}$ and $\mathbf{B}$ in equation (2) is solved using the block-coordinate descent as follows [2], and the details of block-coordinate descent can be found in [28].

$$\mathbf{W} = \mathbf{B}\mathbf{Y}\mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + n\lambda I)^{-1} \tag{6}$$

$$\mathbf{B} = (\gamma\mathbf{P} + \mathbf{W}\mathbf{X}\mathbf{Y})(\gamma\mathbf{Q} + \mathbf{Y}\mathbf{Y}^\top)^{-1} \tag{7}$$

	FreeField1010	Task4 of DCASE2016
#instance	7690	4732
duration (s)	10	4
Total hours	21.3	5.25
Size of vocab.	7724	7

Table 1. Two sound datasets: Freefield1010 and Task4 of DCASE2016.

3. EXPERIMENTAL SETUP

To evaluate the proposed approach (Atag), we conduct experiments on two public datasets, namely FreeField1010 and Task4 of DCASE2016, as described in Table 1.

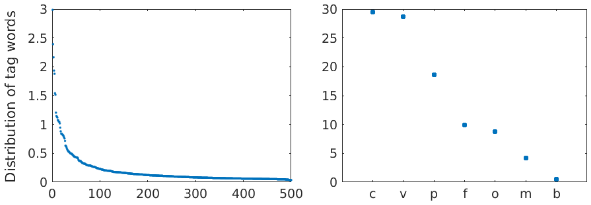

Fig. 2. Distribution of tags.

The audio recordings in Task4 of DCASE2016 [19] are made in a domestic environment. The audio data are 4-second chunks, sampling at 16kHz in mono. For each chunk, multi-label annotations were first obtained from each of the 3 annotators. The annotations are based on a 7 label classes, "*Broadband noise*(b)", "*Child speech*(c)", "*Adult female speech*(f)", "*Adult male speech*(m)", "*Other identifiable sounds*(o)", "*Percussive sounds*(p)", "*Video game/TV*(v)". In our experiments, following the original configuration of Task4 of DCASE2016, we use the same five folds as the test set from the given development dataset, and use the remaining audio recordings for training.

The Freefield1010 dataset [29], originally extracted from Freesound[1], contains 7690 10-second audio files. Each of them is stored in a standardised format at a sampling rate of 44.1kHz in mono. The audio files, provided by users around the world, include many different kinds of natural sound, e.g. bird songs, wind, footsteps, and human voices. Each audio recording generally contains several different sounds and are labelled with rich tags. For simplicity, we reduce the size of tag list to 500 according to the occurrence frequency of tags in the dataset. Figure 2 shows the imbalanced tag distribution over the two datasets. We randomly select 6940 audio files for training, and the rest are for testing.

We pre-process each audio file in the two datasets by segmenting them using a 20ms sliding window with a 10ms hop size, and converting each segment into 24-D MFCCs. For a comparison, we use the GMM method [30, 19] as a baseline, where the number of mixture components is 8 and a binary classifier is built associating with each sound class in the training step. In this work, we treat each selected tag in the tag list as a representation of a distinct sound class, so we built seven GMM classifiers for Task4 of DCASE2016 and 500 classifiers for Freefield1010.

To measure the effectiveness performance of our approach and the baseline, we use equal error rate (EER) for Task4 of DCASE2016 in order to match the evaluation metric originally used and compare with the baseline in [19]. For Freefield1010, we compute precision (P), recall (R), and F1 score ($F1 = 2 * P * R/(P + R)$) on M top-ranked tags according to their scores obtained using the two methods. Due to the relatively poor quality of recorded recordings and the large number of tags in the tag set, not all tags corresponding to an audio recording can be predicted correctly and rank in the top of the list. For the reasons, we are interested in the distribution of the correctly predicted tags ranking in top M. In our experiments, the average number of tags of each audio recording in the test set is 5, so $M = 10$ is large enough for evaluating tag distribution. To measure the performance of efficiency, we record the training time taken when running the two methods, respectively.

4. RESULTS AND ANALYSIS

Table 2. Comparison of EERs using the GMM baseline and Atag on Task4 of DCASE2016.

	fold1	**fold2**	**fold3**	**fold4**	**fold5**	**Avg.**
GMM	0.2205	0.1909	0.1838	0.2182	0.2488	0.2130
Atag	0.2135	0.1787	0.1932	0.2204	0.2239	0.2056

Table 2 shows the values of EER obtained using the baseline and Atag on the five folds. Atag has advantages over the baseline on three folds, and yields 3.4% relative improvement in average on the test set.

Table 3. Comparison of F1 scores (%) of 10 top-rank retrieved tags on Freefield1010.

	Prec.	**Recall**	**F1**	**# correctly retrieved distinct tags in top 10/#distinct tags in test set**
Rand	0.96	2.04	1.31	68/462
GMM	4.70	5.06	4.87	114/462
Atag	5.19	5.67	5.42	129/462

Table 3 shows the F1 scores of the 10 top-ranked tags retrieved by running "Rand", "GMM" and "Atag" on Freefield1010, respectively. For "Rand", we randomly select tags from the tag list of Freefield1010 with respect to the word distributions over this dataset. The F1 scores obtained using Atag

[1] https://www.freesound.org/

is better than the GMM. The rightmost column of the table represents how many distinct tags in the top ten are accurately retrieved. The larger the number is, the better performance it can achieve. As a further comparison, figure 3 shows the F1 scores using different numbers of top ranked tags. Atag achieves the best performance with the threshold of 10, while 13 gives the best result for the baseline.

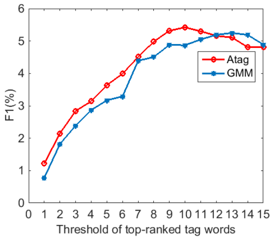

Fig. 3. Comparison of F1 scores using different number of top-ranked tags for GMM and Atag methods.

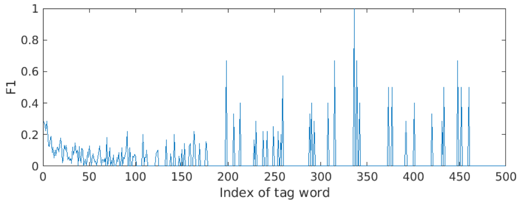

Fig. 4. F1 score of all tags obtained on the test set, ordered by frequency of occurrence (high to low).

As we are especially interested in natural sounds with a large number of tags, In figure 4, we show the F1 scores of all 500 tags obtained on the test dataset. The tags on the left side of x-axis are the words with higher occurrence frequency, and they can thus be retrieved better than the tags with lower occurrence frequency. We also notice that some tags on the right side of x-axis can also reach good F1 scores. Although the number of instances of these sound classes in the training set is not dominant, the sound classes corresponding to these tags might be well defined. In figure 5, we show the number of tags retrieved in the top ten in every test audio recording. We find, in most cases, only one or two tags were retrieved, and many tags do not appear in the top ten tag list. From our point of view, this case is mainly caused by two factors. The first factor is the wide acoustic variations existing in these natural sounds as we mentioned before. The second one is that some of these tags are actually irrelevant to the sound class in the audio file, such as the country' name, address and the brand of microphone. Some of these redundant information possibly works as noise and interfere model optimization finally.

To compare efficiency, table 4 shows the time taken for training. We run Matlab code on Intel Core i7 Processor with 16GB RAM. For Task4 of DCASE2016, a small dataset, the use of Atag is 35 times faster than the baseline. If we process a large dataset, like Freefield1010, the GMM method runs much slower than Atag because it needs to train several hundred GMMs, while Atag needs only to optimize two linear mappings. So from a practical application point of view, the used approach should be an attractive option.

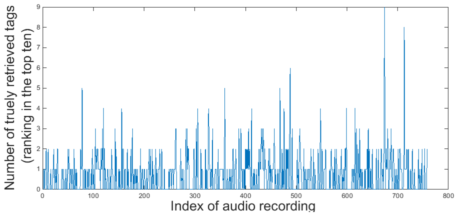

Fig. 5. Number of correctly retrieved tags ranking in the top ten of each audio recording in the test set.

Table 4. Comparison of time taken for training on Freefield1010 and Task4 of DCASE2016.

	Freefield1010	**Task4 of DCASE2016**
Atag	4.3(minutes)	1(minute)
GMM	2460 (minutes)	35 (minutes)

5. CONCLUSION AND FUTURE WORK

In this paper we used a co-regularization approach to jointly optimize classifiers on sound and tag. The use of "corrupted" tags by denoising provided further information for the prediction of true tags. It can enforce the prediction of true tags by adapting the existing labels and making them agree with the tags predicted by the given audio recordings. The use of this approach yielded slight better performances than the GMM baseline on two datasets, and greatly reduced the time taken in training in comparison with the baseline.

We noticed some efforts in Task4 of DCASE2016 using convolutional neural network. We will also consider applying it to some large natural sound resources associated with thousand tags, including many irrelevant tags and some tags not often occurring.

In our future work, we will research four aspects: 1) using more robust low-level audio features instead of only MFCCs. 2) using more complex structure, such as the convolutional neural network, and study how to improve effectiveness and still keep the advantage of linear mapping in efficiency. 3) exploring how to combine the audio and text information by using more semantic information and some natural language processing technologies. 4) integrating multimodal information, i.e. audio, visual and text information, into one framework to make sense of sounds.

6. REFERENCES

[1] M. Wall, "Big data: Are you ready for blast-off?," http://www.bbc.co.uk/news/business-26383058/, 2014, [Online; accessed 4-March-2014].

[2] M. Chen, A. Zhang, and K. Q. Weinberger, "Fast image tagging," in *Proceedings of International Conference of Machine Learning*, 2013, pp. 1274–1283.

[3] N. Venkatesh, M. Subhransu, and R. Manmatha, "Automatic image annotation using deep learning representations," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, 2015, pp. 603–606.

[4] M. K. Mahdi, I. Haroon, and S. Mubarak, "Nmf-knn: Image annotation using weighted multi-view non-negative matrix factorization," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 184–191.

[5] Vikas Sindhwani and Partha Niyogi, "A co-regularized approach to semi-supervised learning with multiple views," in *Proceedings of the ICML Workshop on Learning with Multiple Views*, 2005, pp. 1–6.

[6] Kumar A., P. Rai, and H. III Daum, "A co-regularized approach to semi-supervised learning with multiple views," in *Proceedings of International Conference of Machine Learning*, 2011, pp. 1–9.

[7] U. Brefeld, T. Gartner, T. Scheffer, and S. Wrobel, "Efficient co-regularised least squares regression," in *Proceedings of the International Conference on Machine Learning*, 2006, pp. 137–144.

[8] I. McLoughlin, H. Zhang, Z. P. Xie, Y. Song, and W. Xiao, "Robust sound event classification using deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23(3), pp. 540–552, 2015.

[9] Piczak J. K., "Environment sound classification with convolutional neural networks," in *Proceddings of IEEE International Workshop on Machine Learning for Signal Processing*, 2015, pp. 1–6.

[10] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," *Advances in Neural Information Processing Systems*, pp. 1096–1104, 2009.

[11] B. Elizalde and G. Friedland, "Lost in segmentation: Three approaches for speech/non-speech detection in consumer-produced videos," in *Proceddings of IEEE International Conference on Multimedia and Expo*, 2013, pp. 1–6.

[12] Rybach D., C. Gollan, R. Schlter, and H. Ney, "Audio segmentation for speech recognition using segment features," in *Proceddings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2009, pp. 4197–4200.

[13] H. Meinedo and J. Neto, "A stream-based audio segmentation, classification and clustering pre-processing system for broadcast news using ann models," in *Proceddings of IEEE Interspeech*, 2005, pp. 237–241.

[14] Z. Li and G. D. Guo, "Content-based audio classification and retrieval by support vector machines," *IEEE Transactions on Neural Networks*, pp. 209–215, 2003.

[15] S. Sundaram and S. Naravanan, "Audio retrieval by latent perceptual indexing," in *Proceedings of International Conference on Audio, Speech and Signal Processing*, 2008, pp. 49–52.

[16] J. Foote, M. Cooper, and U. Nam, "Audio retrieval by rhythmic similarity," in *Proceedings of International Conference on Music Information Retrieval*, 2002, pp. 1–6.

[17] T. Bertin-Mahieux, D. Eck, F. Maillet, , and P. Lamere, "A model for predicting social tags from acoustic features on large music databases," *Journal of New Music Research*, vol. 37, pp. 115–135, 2008.

[18] Y. Panagakis, C. Kotropoulos, and R. G. Arce, "Sparse multi-label linear embedding nonegative tensor factorization for automatic music tagging," in *Proceedings of European Signal Processing Conference*, 2010, pp. 492–496.

[19] F. Peter and M. D. Plumbley, "Task4 of DCASE challenge," http://www.cs.tut.fi/sgn/arg/dcase2016/task-audio-tagging/.

[20] A. Makadia, V. Pavlovic, and S Kumar, "Baselines for image annotation," *Journal of Computer Vision*, vol. 90, pp. 88–105, 2010.

[21] J. Tang, R. Hong, S. Yan, T. Chua, G. Qi, and R. Jain, "Image annotation by KNN-sparse graph-based label propagation over noisily tagged web images," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1–15, 2011.

[22] J. Verbeek, M. Guillaumin, T. Mensink, and C Schmid, "Image annotation with tagprop on the MIRFLICKR set," in *Proceedings of ACM Multimedia Information Retrieval*, 2010, pp. 537–546.

[23] L. Wu, R. Jin, and A. K. Jain, "Tag completion for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 716–727, 2013.

[24] Z. Lin, G. Ding, M. Hu, J. Wang, and X. Ye, "Image tag completion via image-specific and tag-specific linear sparse reconstructions," in *Proceedings of CVPR*, 2013, pp. 1618–1625.

[25] Z. Feng, S. Feng, R. Jin, and K. A. Jain, "Image tag completion by noisy matrix recovery," in *Proceedings of ECCV*, 2014, pp. 424–438.

[26] M. Chen, Z. Xu, K. Weinberger, and F. Sha, "Marginalized denoising autoencoders for domain adaptation," in *Proceedings of International Conference of Machine Learning*, 2012, pp. 767–774.

[27] L. Maaten, M. Chen, S. Tyree, and K. Q. Weinberger, "Learning with marginalized corrupted features," in *Proceedings of International Conference of Machine Learning*, 2013, pp. 410–418.

[28] Y. Xu and W Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *Journal on Imaging Sciences*, vol. 6(3), pp. 1758–1798, 2013.

[29] D. Stowell and M. D. Plumbley, "An open dataset for research on audio field recording archives: freefield1010," http://arxiv.org/abs/1309.5275/.

[30] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval od music and sound effects," *IEEE Transactions on Audio Speech and Language Processing*, vol. 16, pp. 1–9, 2008.