LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-650594

AUTOMATIC IMAGE ANNOTATION USING INVERSE MAPS FROM SEMANTIC EMBEDDINGS

J. J. Thiagarajan, K. N. Ramamurthy, P. S. Sattigeri, P. T. Bremer, A. Spanias

February 25, 2014

IEEE International Conference on Image Processing
Paris, France
October 27, 2014 through October 30, 2014

Disclaimer

AUTOMATIC IMAGE ANNOTATION USING INVERSE MAPS FROM SEMANTIC EMBEDDINGS

J.J. Thiagarajan[1], *K.N. Ramamurthy*[2], *P. Sattigeri*[3], *P.T. Bremer*[1], *and A. Spanias*[3]

[1]Lawrence Livermore National Laboratory
[2]IBM Thomas J. Watson Research Center
[3]SenSIP Center, School of ECEE, Arizona State University

ABSTRACT

Human annotation in large scale image databases is time-consuming and error-prone. Since it is very hard to mine image databases using just visual features or semantic descriptors, it is common to transform the image features into a semantically meaningful space for tag prediction. In this paper, we propose to perform image annotation in a semantic space inferred based on sparse representations. By constructing a semantic embedding for the visual features, that is constrained to be close to the tag embedding, we show that a robust inverse map can be used to predict the tags. Experiments using standard datasets show the effectiveness of the proposed approach in automatic image annotation when compared to existing methods.

Index Terms— image annotation, sparse coding, embedding, inverse map, RBF interpolation.

1. INTRODUCTION

Textual information or tags can be useful meta-data for images. In large scale image retrieval systems, it is typical to present a textual query to retrieve semantically relevant images. Since a semantic concept can manifest into a wide range of visual representations, it is often dificult to mine through an image database only based on visual features or the tags. For example, the same set of tags {*Child*, *Car*}, can describe images of a child inside a car, or a child playing with a toy car (Figure 1). This illustrates the impreciseness and incompleteness of available tags. In such cases, the image content can be used to resolve the ambiguity in the concepts associated with images and discover additional descriptive tags. Hence, the goal of automatic image annotation is to predict new tags, and possibly refine existing noisy tags, based on information from visually similar images.

The problem of image annotation has been studied extensively in the recent years. A popular class of algorithms attempt to learn appropriate image classifiers to predict concepts and keywords [1, 2, 3]. Learning a regressor or a classifer to predict tags using the visual features is highly ill-posed. This is because a mapping from features to semantic tags may not exist, and need not be smooth even if it exists. As a result, approaches that infer correlations between visual featurs and textual descriptors have been found to be more effective [4, 5, 6, 7].

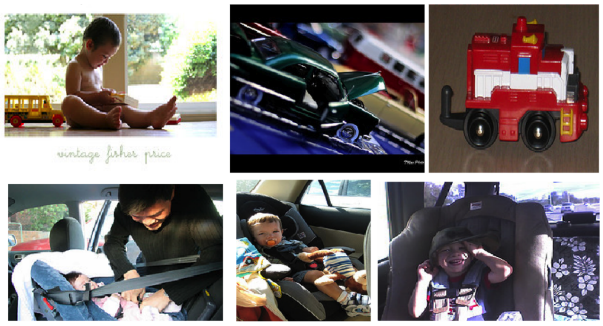

Fig. 1. Sample images that can be described by the same set of tags {*Child, Car*}. The large variability in the image content makes it hard to retrieve semantically relevant images just based on the tags or visual features.

An important challenge in designing image annotation systems is to effectively measure semantic similarities between two images with multiple labels. This requires a clear understanding of the relation between the spaces of visual features and semantic descriptors. By transforming visual features into a semantically meaningful space, one can infer similarities between images more robustly. In [8], the authors map both visual features and tags to the same latent space using canonical correlation analysis and incorporate high-level image semantics. Based on the intuition that images with similar features might have similar tags, the authors in [9] create an embedding for the features such that the relations between their corresponding tag vectors are satisfied.

In this paper, we develop a new approach to compute semantically relevant embeddings for visual features, and also present an algorithm for automatic annotation. We propose to use reference-based features [10] to describe the visual content, and employ sparse representations to infer relationship between images, taking into account both features and tags. Using the set of joint sparse codes, we propose to compute

feature embeddings that are semantically meaningful. Finally, we build an inverse map to predict the image tags from its low-dimensional feature embedding. We evaluate the proposed method on two different datasets and demonstrate its effectiveness in discovering the semantic concepts.

2. FEATURE EXTRACTION

In order to allow for improved semantic comparison between different images, we propose to employ reference-based features in image annotation. While reference-based feature extraction has been successfully used in object recognition [10], it is a natural candidate for use in multi-label annotation. Similar to transfer learning approaches [11], it is assumed that we have access to a reference set of annotated images, different from the images used for training. It is important to note that the collection of labels in the reference set can be completely different from that of the training data. The reference set comprises groups of annotated images, in which each group contains images that share a label. In this feature extraction method, we compute the average similarity of an image feature to each group in the reference set. As a result, we represent an image using confidence measures for the relevance of every label in the reference set to the image feature.

Given an image I, we extract dense SIFT descriptors from overlapping patches of size 16×16. In order to aggregate the descriptors, we use the procedure in [12] to construct sparse coding based spatial pyramid (ScSPM) features. We aggregate the sparse codes for all descriptors in a spatial region by max-pooling [12]. We construct a spatial pyramid by aggregating the sparse codes at multiple spatial scales. Let us denote the ScSPM features for the T training images as $\{\mathbf{h}_i\}_{i=1}^{T}$. Using a similar procedure, we build the ScSPM features for all images in the reference set. We denote the features in a reference group k by $\{\mathbf{r}_n^k\}_{n=1}^{n_k}$, where n_k denotes the total number images in that group. For each training image i, we obtain the similarity between its feature and that of a reference-set image using the measure proposed in [10]

$$S(\mathbf{h}_i, \mathbf{r}_n^k) = 1 - \frac{\gamma\left(\frac{k}{2}, \frac{d(\mathbf{h}_i, \mathbf{r}_n^k)}{2}\right)}{\Gamma(\frac{t}{2})}, \tag{1}$$

where $d(\mathbf{h}_i, \mathbf{r}_n^k)$ denotes the χ^2 distance between the features, $\gamma(.)$ is the lower incomplete gamma function, Γ denotes the gamma function and t is a positive integer that specifies the number of degrees of freedom. The second term in the expression is the cumulative distribution for chi-squared distribution. For a reference class k, we use the average similarity of the training feature with respect to all features in that class as the k^{th} dimension for the feature vector $\mathbf{x}_i$

$$\mathbf{x}_i(k) = \frac{1}{n_k} \sum_{n=1}^{n_k} S(\mathbf{h}_i, \mathbf{r}_k^n). \tag{2}$$

Following this, we obtain the reference-based feature by normalizing $\mathbf{x}_i$ to unit ℓ_2 norm. Features corresponding to all training images are computed and stored in the matrix $\mathbf{X} \in \mathbb{R}^{R \times T}$, where R denotes the number of reference classes.

3. LEARNING SEMANTIC EMBEDDINGS

In order to perform tag prediction, we propose to explore the low-dimensional structure of the feature and tag spaces, and thereby infer the relationship between them. Though features (or tags) lie in a high-dimensional space, inferring the underlying low-dimensional structures will reveal only essential information with little noise, and hence result in improved tag prediction. We exploit the correlations between the features and tags by computing joint sparse codes for the feature-tag matrix $\mathbf{D} = [\mathbf{X}^T \; \gamma\mathbf{B}^T]^T$, where $\mathbf{B} \in \mathbb{R}^{L \times T}$ is the matrix of tag vectors, and γ is the scaling factor used to balance the total energy of feature and tag spaces. We assume that the features and tags are clustered along subspaces, and hence this structure can be discovered using sparse coding on examples as

$$\min_{\mathbf{A}} \|\mathbf{D} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{i=1}^{T} \|\mathbf{a}_i\|_1 \text{ s.t. } a_{ii} = 0, \forall i. \tag{3}$$

Here $\mathbf{A} \in \mathbb{R}^{T \times T}$ is the matrix of sparse coefficients, and the constraint ensures that a sample will not be used in its own representation.

We propose to compute low-dimensional embeddings for the features and the tags using the relationships encoded by $\mathbf{A}$. Furthermore, we will also constrain the features to be projected in the neighborhood of the tag embedding. This will ensure that the resulting space is semantically meaningful and can be used for effective tag prediction. The joint optimization problem for computing the feature and the tag embeddings is posed as

$$\begin{aligned} \min_{\mathbf{P},\mathbf{Q},\{\boldsymbol{\delta}_i\}_{i=1}^{T}} & \sum_{i=1}^{T} \|\mathbf{P}^T\mathbf{x}_i - \sum_j a_{ij}\mathbf{P}^T\mathbf{x}_i\|_2^2 \\ +\lambda_1\|\mathbf{Q}^T\mathbf{b}_i - \textstyle\sum_j a_{ij}\mathbf{Q}^T\mathbf{b}_i\|_2^2 & + \lambda_2\|\mathbf{P}^T\mathbf{x}_i - \mathbf{Q}^T\mathbf{B}\boldsymbol{\delta}_i\|_2^2 \\ +\lambda_3\|\mathbf{W}_i\boldsymbol{\delta}_i\|_1 \text{ s.t. } & \mathbf{P}^T\mathbf{P} = \mathbf{I}, \mathbf{Q}^T\mathbf{Q} = \mathbf{I}. \end{aligned} \tag{4}$$

The matrices $\mathbf{P} \in \mathbb{R}^{R \times d}$ and $\mathbf{Q} \in \mathbb{R}^{S \times d}$ where $d < R, d < S$ define the low-dimensional embeddings of the features and the tags respectively. The first two terms preserve the sparse coding relationships for both features and tags in the low-dimensional space. The next two terms constrain that the embedded features be close to a local sparse linear combination of the embedded tags. The weight matrix $\mathbf{W}_i$ ensures that the embedded tags closest to $\mathbf{P}^T\mathbf{x}_i$ are chosen when computing the sparse code $\boldsymbol{\delta}_i$. This problem is jointly non-convex over $\mathbf{P}, \mathbf{Q}, \{\boldsymbol{\delta}_i\}_{i=1}^{T}$, and hence we choose to minimize it in an alternating manner over each of the variables.

Table 1. Precision-Recall rates for image annotation on the Corel-5K dataset.

Algorithm	Avg. Precision	Avg. Recall
CMRM [6]	0.1	0.09
CRM [7]	0.16	0.19
CRM-Rect [7]	0.22	0.23
MBRM [5]	0.24	0.25
SML [13]	0.23	0.29
JEC [14]	0.27	0.32
MSC [9]	0.25	0.32
GS [15]	0.3	0.33
Proposed	**0.37**	**0.4**

Assuming that $\mathbf{Q}$ and $\{\boldsymbol{\delta}_i\}_{i=1}^T$ are known, we solve for $\mathbf{P}$ using

$$\min_{\mathbf{P}} \text{Tr}(\mathbf{P}^T\mathbf{X}\mathbf{M}\mathbf{X}^T\mathbf{P}) - 2\lambda_2\text{Tr}(\mathbf{P}^T\mathbf{X}\boldsymbol{\Delta}^T\mathbf{B}^T\mathbf{Q}) \quad \text{s.t. } \mathbf{P}^T\mathbf{P} = \mathbf{I}. \tag{5}$$

where $\mathbf{M} = (\mathbf{I}-\mathbf{A})^T(\mathbf{I}-\mathbf{A}) + \lambda_2\mathbf{I}$, $\boldsymbol{\Delta} = [\boldsymbol{\delta}_1\boldsymbol{\delta}_2\ldots\boldsymbol{\delta}_T]$, and Tr(.) is the trace operator. In (5), the feasible set is $\{\mathbf{P} : \mathbf{P}^T\mathbf{P} = \mathbf{I}\}$ and hence $\mathbf{P}$ lies in a Steifel manifold [16]. This general problem of optimization over the manifold is solved using the curvilinear search method with Barzilai-Borwein (BB) steps as described in [17],[18]. In the second alternating step, we optimize over $\mathbf{Q}$, using the latest values for the other two variables. This can be written as

$$\min_{\mathbf{Q}} \text{Tr}(\mathbf{Q}^T\mathbf{B}\boldsymbol{\Delta}\mathbf{N}\boldsymbol{\Delta}^T\mathbf{B}^T\mathbf{Q}) - 2\lambda_2\text{Tr}(\mathbf{Q}^T\mathbf{B}\boldsymbol{\Delta}\mathbf{X}^T\mathbf{P}) \quad \text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}. \tag{6}$$

where $\mathbf{N} = \lambda_2(\mathbf{I}-\mathbf{A})^T(\mathbf{I}-\mathbf{A}) + \mathbf{I}$, and hence can be solved in a similar manner as (5). The third step is to solve for the sparse codes $\{\boldsymbol{\delta}_i\}$. Each sparse code can be solved as,

$$\min_{\boldsymbol{\delta}_i} \|\mathbf{P}^T\mathbf{x}_i - \mathbf{Q}^T\mathbf{B}\boldsymbol{\delta}_i\|_2^2 + \frac{\lambda_3}{\lambda_2}\|\mathbf{W}_i\boldsymbol{\delta}_i\|_1 \tag{7}$$

where the diagonal weight matrix $\mathbf{W}_i$ imposes a penalty that encourages the selection of the embedded tags, $\{\mathbf{Q}^T\mathbf{b}_j\}_{j=1}^T$, that are close to the embedded feature, $\mathbf{P}^T\mathbf{x}_i$. In practice, we simplify this problem by choosing the k nearest neighbors of $\mathbf{P}^T\mathbf{x}_i$ from $\{\mathbf{Q}^T\mathbf{b}_j\}_{j=1}^T$, which is equivalent to setting the penalties for neighbors as 1, and non-neighbors as ∞ in $\mathbf{W}_i$. This is then solved as a least squares problem on the chosen neighbors. In this work, we experimentally determined that $k = 5$ provides a good performance.

4. TAG PREDICTION USING INVERSE MAPS

Using the semantic embedding, we project the image features onto the neighborhood of the low-dimensional tag embedding. As a result, $\mathbf{P}^T\mathbf{X}$ can be considered to be lying close to some smooth embedding of the tag space itself. Predicting tags amounts to computing an inverse map from the $d-$dimensional space to the space of tags. This is very similar to the problem of inverting embeddings obtained using non-linear dimensionality reduction in the manifold learning literature. Though different forms of mapping can be considered, it has been found that inverse maps obtained using the natural cubic radial basis function (RBF) kernel is very effective in reconstructing the underlying structure of the manifold [19]. Note that, the effectiveness of this inverse mapping depends strongly on the choice of the dimension d, smoothness of the embedding, and the availability of enough samples. Hence, we propose to learn a natural cubic interpolant in the semantic space using $\mathbf{Y} = \mathbf{P}^T\mathbf{X}$ to predict tags.

For any novel sample $\mathbf{y}$, we can construct the natural cubic interpolant for each dimension of the tag vector as follows:

$$\sum_{j=1}^{T} c_j(\mathbf{y})k(\mathbf{y},\mathbf{y}_j) + \mathbf{y}^T\boldsymbol{\alpha}_1 + \alpha_0 \quad \text{subj. to } \sum_{j=1}^{T} c_j = 0, \sum_{j=1}^{T} c_j\mathbf{y}_j = 0, \tag{8}$$

where $k(\mathbf{y},\mathbf{y}_j) = \|\mathbf{y}-\mathbf{y}_j\|_2^3$, α_0 is a constant term, and $\boldsymbol{\alpha}_1 \in \mathbb{R}^d$. Using the set of training samples, we can estimate the interpolation parameters by solving the following system of linear equations.

$$\begin{bmatrix}\mathbf{K} & \mathbf{Z}\\ \mathbf{Z}^T & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{C}\\ \boldsymbol{\alpha}\end{bmatrix} = \begin{bmatrix}\mathbf{B}\\ \mathbf{0}\end{bmatrix}, \tag{9}$$

where $\mathbf{K} \in \mathbb{R}^{T\times T}$ is the kernel matrix, $\mathbf{Z} = [\mathbf{1}\ \mathbf{Y}]^T$, $\mathbf{C} \in T \times L$ is interpolation weight matrix, $\boldsymbol{\alpha}_0 \in \mathbb{R}^{1\times L}$, $\underline{\boldsymbol{\alpha}}_1 \in \mathbb{R}^{T\times L}$, and $\underline{\boldsymbol{\alpha}} = [\boldsymbol{\alpha}_0^T\ \underline{\boldsymbol{\alpha}}_1^T]^T$. To annotate a test image I_t, we need to extract the reference based feature and project it onto the semantic space $\mathbf{y}_t = \mathbf{P}^T\mathbf{x}_t$. Finally, its tag vector can be predicted with the inverse map using (8).

Table 2. Precision-Recall rates for image annotation on the IAPR TC-12 dataset.

Algorithm	Avg. Precision	Avg. Recall
MBRM [5]	0.24	0.23
JEC [14]	0.28	0.29
Lasso [14]	0.28	0.29
GS [15]	0.32	0.29
Proposed	**0.39**	**0.34**

5. PERFORMANCE EVALUATION

We studied the performance of the proposed algorithm using two different datasets: (a) Corel-5K [20], and (b) IAPR

Table 3. Annotation results obtained for sample test images from the IAPR TC-12 dataset. In each case, the ground-truth human annotation is also included.

					
Human Annotation	desert, middle, rock, tourist	airplane, desert, middle, mountain	cloud, fence, hill, man, meadow, statue, wall	bridge, building, meadow, river, sky, skyscraper, tree	bay, cloud, coast, grass, hill, sea, sky
Proposed Algorithm	desert, rock, people, sky, group	airplane, mountain, sky, cloud, building	statue, fence, mountain, cloud, grass	building, sky, city, grass, water	water, cloud, grass, hill, sky

TC-12 [14]. In both these datasets, each image is characterized by multiple tags, and performance evaluation is carried out on a test set different from the images used during training. Commonly used performance metrics are the precision and recall values for each tag in the vocabulary. Precision of a label is measured as the ratio between the number of correctly annotated images and the total number of images annotated with that tag by the algorithm. Recall is defined as the ratio between the number of correctly annotated images and the number of images that have the said label in the ground truth annotation. Similar to the procedure followed in [9], we choose the top 5 tags, with the largest propagation scores in the predicted tag vector, as the annotation for a test image.

Parameter Setting: The size of the codebook for computing the ScSPM feature was fixed at 1024, and the sparse codes were aggregated using max-pooling in 3 spatial levels. The resulting ScSPM feature vector was of size $21,504$. In order to compute the sparse codes for both features and tags in (3), the sparstiy penalty was fixed at 0.3. In all our simulations, the number of projection directions d was fixed at 100.

Corel-5K Dataset: This is a very commonly used comparative dataset for image annotation. There are 5000 images in total, and each image is annotated with 1 to 5 keywords. We used 4500 images for training the algorithm, and evaluated the performance using the rest. The total number of keywords in the vocabulary is 260. For building the referece set, we used the IAPR images and grouped them into categories based on their tags. Note that, we ignored tags which had the same set of images in them. As a result, the feature vector for each image, computed using the procedure described in Section 2 were of size 259×1.

IAPR TC-12 Dataset: This dataset is a collection of about $20,000$ natural images including people, animals, cities, landscape etc. Unlike the Corel-5K dataset, the images in this dataset are accompanied by free-flowing text captions, and hence it is common to extract nouns from these captions to build the tags. We used the ground-truth image annotation generated by the authors in [14]. The total of number of labels is 291, where each image is annotated by about 4.7 tags on average. Following the standard evaluation procedure, we used $17,500$ images for training and the rest for testing. Since the Corel-5K dataset was used as the reference set, the feature vectors were of size 260×1.

Results: Tables 1 and 2 show the average precision and recall values obtained with the two datasets. For comparison, we have included the performance of other existing methods for image annotation. The proposed algorithm provided superior annotation results in both cases and outperformed existing schemes. Table 3 shows a sample set of images from the IAPR TC-12 dataset along with their predicted tags. Our algorithm was able to infer meaningful descriptors for these images such as the objects present in them, different components of the background and in some cases abstract concepts that the human annotation does not reveal. For example, in the fourth image (Table 3), though the proposed method missed to identify the tags *skyscraper* and *bridge*, it has inferred the keyword *city* which is a higher-level description of the image content. The results demonstrate that it is important to explore correlations in both features and tags to create semantically meaningful representations for images. Furthermore, by constructing appropriate embeddings, tag prediction can be performed using inverse maps efficiently. It will be interesting to investigate the use of the proposed algorithm in building higher-level abstract concepts based on tags inferred for the different local regions in an image.

6. REFERENCES

[1] G. Carneiro, A.B. Chan, P.J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 3, pp. 394–410, March 2007.

[2] E. Chang, Kingshy Goh, G. Sychay, and G. Wu, "Cbsa: content-based soft annotation for multimodal image retrieval using bayes point machines," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 1, pp. 26–38, Jan 2003.

[3] C. Cusano, G. Ciocca, and R. Schettini, "Image annotation using SVM," *Proceedings of SPIE*, vol. 5304, pp. 330–338, 2004.

[4] David M. Blei and Michael I. Jordan, "Modeling annotated data," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 2003, pp. 127–134.

[5] SL Feng, Raghavan Manmatha, and Victor Lavrenko, "Multiple bernoulli relevance models for image and video annotation," in *Computer Vision and Pattern Recognition, Proceedings of the 2004 IEEE Computer Society Conference on*, 2004, vol. 2.

[6] Jiwoon Jeon, Victor Lavrenko, and Raghavan Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 2003, pp. 119–126.

[7] Victor Lavrenko, R Manmatha, and Jiwoon Jeon, "A model for learning the semantics of pictures," *Advances in neural information processing systems*, 2003.

[8] Y. Gong *et al.*, "A multi-view embedding space for modeling internet images, tags, and their semantics," *arXiv preprint arXiv:1212.4522*, 2012.

[9] C. Wang *et al.*, "Multi-label sparse coding for automatic image annotation," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 1643–1650.

[10] Qun Li, Honggang Zhang, Jun Guo, Bir Bhanu, and Le An, "Reference-based scheme combined with k-svd for scene image categorization.," *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 67–70, 2013.

[11] Sinno Jialin Pan and Qiang Yang, "A survey on transfer learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, Oct 2010.

[12] J. Yang, K. Yu, Y. Gong and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. of IEEE CVPR*, Jun. 2009.

[13] Gustavo Carneiro, Antoni B Chan, Pedro J Moreno, and Nuno Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 3, pp. 394–410, 2007.

[14] Ameesh Makadia, Vladimir Pavlovic, and Sanjiv Kumar, "A new baseline for image annotation," *Computer Vision–ECCV 2008*, pp. 316–329, 2008.

[15] S. Zhang *et al.*, "Automatic image annotation using group sparsity," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 3312–3319.

[16] William M Boothby, *An Introduction to Differentiable Manifolds and Riemannian Geometry*, Academic Press, 2003.

[17] Zaiwen Wen and Wotao Yin, "A feasible method for optimization with orthogonality constraints," *Mathematical Programming*, pp. 1–38, 2013.

[18] Zaiwen Wen and Wotao Yin, "Matlab software for optimization with orthogonality constraints," *Available online at http://optman.blogs.rice.edu/*.

[19] Nathan D Monnig, Bengt Fornberg, and Francois G Meyer, "Inverting non-linear dimensionality reduction with scale-free radial basis interpolation," *arXiv preprint arXiv:1305.0258*, 2013.

[20] P. Duygulu *et al.*, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in *Proceedings of the 7th European Conference on Computer Vision-Part IV*, 2002, ECCV '02, pp. 97–112.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.