Locality Discriminative Coding for Image Classification

Xiaoshan Yang
National Lab of Pattern Recognition
Institute of Automation
Chinese Academy of Sciences
Beijing, China
xiaoshan.yang@nlpr.ia.ac.cn

Tianzhu Zhang
National Lab of Pattern Recognition
Institute of Automation
Chinese Academy of Sciences
Beijing, China
tzzhang10@gmail.com

Changsheng Xu
National Lab of Pattern Recognition
Institute of Automation
Chinese Academy of Sciences
Beijing, China
csxu@nlpr.ia.ac.cn

ABSTRACT

The Bag-of-Words (BOW) based methods are widely used in image classification. However, huge number of visual information is omitted inevitably in the quantization step of the BOW. Recently, NBNN and its improved methods like Local NBNN were proposed to solve this problem. Nevertheless, these methods do not perform better than the state-of-the-art BOW based methods. In this paper, based on the advantages of BOW and Local NBNN, we introduce a novel locality discriminative coding (LDC) method. We convert each low level local feature, such as SIFT, into code vector using the **Local Feature-to-Class** distance other than by k-means quantization. Extensive experimental results on 4 challenging benchmark datasets show that our LDC method outperforms 6 state-of-the-art image classification methods (3 based on NBNN, 3 based on BOW).

Categories and Subject Descriptors

I.2.6 [**Learning**]: Sparse learning; I.4.10 [**Image Representation**]: Miscellaneous; I.5 [**Pattern Recognition**]: Computer Vision

General Terms

Algorithms, Experimentation, Performance

Keywords

Bag-of-Words, Feature Coding, Discriminative

1. INTRODUCTION

Image classification is one of the most important and challenging research tasks in computer vision. The improvement on image classification can also benefit other useful applications, such as image search and retrieval. Recently, the Bag-of-Words (BOW) based methods [13, 3] are popularly used in image classification [8, 6]. The conventional BOW pipeline consists of five stages: feature extraction, codebook design, feature coding, feature pooling, classifier construction. Despite remarkable progress has been made in these five stages, there exists much more room for improvement especially in the feature coding step.

Here, we simply review several popular BOW based feature coding methods. Lazebnik *et al.* [8] adopted spatial pyramid matching (SPM) to consider global geometric correspondence among local features and a voting scheme for coding, which is simple yet highly sensitive to reconstruction errors induced by the codebook. Yang *et al.*[17] proposed a method called ScSPM which introduces sparse regularization to the soft-assignment coding method. Thus, competitive image classification results were obtained just by using linear SVM. Wang *et al.*[16] further improved ScSPM with a locality constraint which leads to an analytical solution to the coding problem and a fast approximated solving method. Recently, Liu *et al.*[10] proposed localized soft-assignment coding(LSC) based on traditional soft-assignment coding method by adding locality constraint on the distance function. Their motivation is similar to the salient coding method proposed by Huang *et al.*[7].

The basic idea of the above BOW based feature coding methods is borrowed from text retrieval [13, 3]. As we know, word is the smallest semantic unit in text. Thus, the quantized vector which represents the frequency of the extracted meaningful semantic words in the documents retains almost all semantic information. However, in image classification, two problems can not be omitted. **(1)** We can not guarantee that the constructed finite visual words have the independent semantic information like words in text retrieval task. **(2)** In BOW [1, 2], the k-means quantization results in a substantial loss of discriminative power for the local visual features. As pointed out by Boiman *et al.* [2], most of the densely sampled local features such as SIFT comprise of simple edges and corners which are actually least informative for classification. In contrast, the most informative local features are always rare in the dataset. Any clustering of the feature space into a small number of visual words will inevitably lead to a high quantization error, because the clustering centers (visual words) are determined by the most frequent data points which are least informative.

To solve the above two problems, a new non-parametric method named Naive-Bayes Nearest-Neighbor(NBNN) was proposed in [2]. In NBNN, the local features were retained in their original form without quantization. As a result, it takes the best use of the full discriminative power of the local

ICIMCS '13, Aug. 17-19, 2013, Huangshan, Anhui, China

features. However, this method is extremely slow and does not perform better than state-of-the-art BOW based methods. To improve the effectiveness and efficiency of NBNN, many relevant methods were proposed. Tuytelaars *et al.*[14] proposed NBNN Kernel which can be seemed as constructing a kernel function using NBNN. Sancho *et al.*[11] proposed Local NBNN method to improve NBNN by ignoring the distances to classes far from the query feature, which also leads to a speeding up to the original NBNN method. Nevertheless, the classification accuracies of the NBNN based methods [2, 14, 11] are still much lower than the improved BOW methods, such as localized soft-assignment coding (LSC) [10].

Inspired by the previous BOW and NBNN based methods, we propose a Locality Discriminative Coding (LDC) method which has two advantages for feature coding. **(1)** Without k-means quantization, we adopt the distance between a local feature and its nearest neighbor in each class for feature coding. Thus, we obtain encoded feature vectors with more discriminative power compared with the BOW based methods. Fig. 1 shows an illustration of the difference between our discriminative coding method and BOW based coding method. **(2)** Since our coding method is based on Local NBNN, the locality and saliency properties can be transferred seamlessly from Local NBNN into our coding method. After feature coding, average pooling and SPM are adopted to obtain the image level representation. Finally, the linear SVM [5] is applied to perform classification. Extensive experimental results on several popular benchmarks show that our LDC outperforms both BOW and NBNN based methods.

2. RELATED WORK

In this section, we review several popularly used BOW [3, 15, 17, 16, 10] based coding methods and NBNN [2] method for image classification. Let $\vec{\mathbf{b}}_i$ ($\vec{\mathbf{b}}_i \in \Re^d$) denote a visual word, where d is the dimensionality of the local feature. Matrix $\mathbf{B} = \left[\vec{\mathbf{b}}_1, \vec{\mathbf{b}}_2, \cdots, \vec{\mathbf{b}}_m\right]$ denotes a visual codebook consists of m visual words. Let $\vec{\mathbf{x}}_i$ ($\vec{\mathbf{x}}_i \in \Re^d$) be the i^{th} local feature in an image $\mathbf{X} = [\vec{\mathbf{x}}_1, \vec{\mathbf{x}}_2, \cdots, \vec{\mathbf{x}}_n]$. Let $\vec{\mathbf{v}}_i$ ($\vec{\mathbf{v}}_i \in \Re^m$) be the code vector of $\vec{\mathbf{x}}_i$, with $\mathbf{v_{ij}}$ being the coefficient respected to word $\vec{\mathbf{b}}_j$.

Hard-assignment coding: In original Bag-of-Words method [3], the simplest hard-assignment or vector quantization was used to construct a bag of keypoints(original name of visual words). The main idea is to count the number of local features assigned to each visual word. Given a query local feature $\vec{\mathbf{x}}_i$, the formal expression can be written as following.

$$\mathbf{v_{ij}} = \begin{cases} 1 & \textbf{if} \quad j = \underset{j=1,\ldots,m}{\arg\min} \left\|\vec{\mathbf{x}}_i - \vec{\mathbf{b}}_j\right\|_2^2 \\ 0 & \textbf{otherwise} \end{cases} \tag{1}$$

Soft-assignment coding: A local feature $\vec{\mathbf{x}}_i$ is assigned to just a single visual word by Hard-assignment coding while Soft-assignment coding assigns it to all visual words in the codebook $\mathbf{B}$ [15] with different weights.

$$\mathbf{v_{ij}} = \frac{\exp(-\alpha \left\|\vec{\mathbf{x}}_i - \vec{\mathbf{b}}_j\right\|_2^2)}{\sum_{k=1}^{m} \exp(-\alpha \left\|\vec{\mathbf{x}}_i - \vec{\mathbf{b}}_k\right\|_2^2)} \tag{2}$$

Sparse coding: In this method [17], an optimization problem with sparse regularization constraint needs to be solved.

$$\vec{\mathbf{v}}_i = \underset{\vec{\mathbf{v}}_i \in \Re^m}{\arg\min} \|\vec{\mathbf{x}}_i - \mathbf{B}\vec{\mathbf{v}}_i\|_2^2 + \lambda \|\vec{\mathbf{v}}_i\|_1 \tag{3}$$

Locality constrained linear coding: In this method [16], a locality constraint is introduced by adding Euclidean distance weight factor $\mathbf{d_i} = exp\left(dist\left(\vec{\mathbf{x}}_i, \mathbf{B}\right)/\delta\right)$ in Eq.(3). Symbol $\odot$ represents element-wise multiplication.

$$\begin{aligned} \vec{\mathbf{v}}_i &= \underset{\vec{\mathbf{v}}_i \in \Re^m}{\arg\min} \|\vec{\mathbf{x}}_i - \mathbf{B}\vec{\mathbf{v}}_i\|_2^2 + \lambda \left\|\mathbf{d_i} \odot \vec{\mathbf{v}}_i\right\|_2^2 \\ s.t. \quad & \mathbf{1}^T \vec{\mathbf{v}}_i = 1 \end{aligned} \tag{4}$$

Localized soft-assignment coding: As proposed by Liu *et al.*[10], localized soft-assignment coding method mainly focuses on the k neighborhood visual words for each local feature $\vec{\mathbf{x}}_i$. Here $\mathbf{N}_k\left(\vec{\mathbf{x}}_i\right)$ is the k-nearest neighbors of $\vec{\mathbf{x}}_i$.

$$\mathbf{v_{ij}} = \frac{\exp(-\alpha \left\|\vec{\mathbf{x}}_i - \vec{\mathbf{b}}_j\right\|_2^2)}{\sum_{k=1}^{n} \exp(-\alpha \left\|\vec{\mathbf{x}}_i - \vec{\mathbf{b}}_k\right\|_2^2)}, \vec{\mathbf{b}}_k \in \mathbf{N}_k\left(\vec{\mathbf{x}}_i\right) \tag{5}$$

NBNN: NBNN [2] is a nonparametric image classifier that achieves impressive accuracy by exploiting 'Image-to-Class' distances and by avoiding quantization of local features. In the implementation, the 'Image-to-Class' distances are adopted to infer the class label of an query image. Here, $NN_C(\vec{\mathbf{x}}_i)$ is the nearest neighbors of $\vec{\mathbf{x}}_i$ in the local feature subset which is consisted of all training images from class C.

$$C^* = \underset{C}{\arg\min} \sum_{i=1}^{n} \|\vec{\mathbf{x}}_i - NN_C(\vec{\mathbf{x}}_i)\| \tag{6}$$

3. OUR ALGORITHM

In this section, we give the detail of our LDC for image classification, and explain why it can solve the problems of both BOW and NBNN based methods.

3.1 Our Formulation

The BOW based methods as shown in Section 2 can be summarized as computing a probability of local feature to the visual words. Each feature coding vector $\vec{\mathbf{v}}_i$ that is a semantic description of $\vec{\mathbf{x}}_i$ can be computed as

$$\mathbf{v_{ij}} = P\left(\vec{\mathbf{b}}_j | \vec{\mathbf{x}}_i\right) \propto s\left(\vec{\mathbf{x}}_i, \vec{\mathbf{b}}_j\right) \qquad j = 1, ..., m \tag{7}$$

Here, $s\left(\vec{\mathbf{x}}_i, \vec{\mathbf{b}}_j\right)$ is the similarity measurement of $\vec{\mathbf{x}}_i$ and $\vec{\mathbf{b}}_j$, which can be estimated by any coding methods as shown in Section 2. This is a more generalized version of the probability expression for the traditional coding methods compared with the probability expression in [10]. Therefore, we can see that the BOW based methods only use the clustered centers to classify a query image. This means that we compute the distances among thousands of local features to only get the cluster centers, and the huge amount of distance information computed in the process of clustering are discarded. Inspired by SVM where the hyperplane is decided only by supporter vectors, we may think that a small subset of the local features is enough. The problem is that we can not guarantee the clustered centers are exactly distributed around the hyperplane in the local feature space.

Based on the above analysis, we believe that taking the best use of the distances among local features is more important than just finding improved method for clustering visual

words in the local feature space. Inspired by NBNN image classification method [2], we propose a novel feature coding method. We compute the posteriori probability of a local feature to each category rather than to the each visual word in local feature space.

$$\mathbf{v_{ij}} = P\left(C_j|\vec{\mathbf{x}}_i\right) = P\left(\vec{\mathbf{x}}_i|C_j\right)\frac{P\left(C_j\right)}{P\left(\vec{\mathbf{x}}_i\right)} \quad j = 1, ..., m_c \tag{8}$$

Here C_j denotes the j^{th} class of the image classification task, m_c is the number of image classes. When the class prior $p\left(C_j\right)$ is uniform for different classes, computing the feature descriptor $\vec{\mathbf{v}}_i$ can be deduced to compute the posterior probability $P\left(\vec{\mathbf{x}}_i|C_j\right)$.

We use the Parzen density estimation $\hat{P}\left(\vec{\mathbf{x}}_i|C_j\right)$ to approximate $P\left(\vec{\mathbf{x}}_i|C_j\right)$ as [4, 2]

$$\hat{P}\left(\vec{\mathbf{x}}_i|C_j\right) = \frac{1}{L}\sum_{k=1}^{L} K\left(\vec{\mathbf{x}}_i - \vec{\mathbf{x}}_k^{C_j}\right) \tag{9}$$

where $\{\vec{\mathbf{x}}_1^{C_j}, ..., \vec{\mathbf{x}}_L^{C_j}\}$ denote all the local features that belong to class C_j in the training image set, and $K(.)$ can be a non-negative function, such as Gaussian function, to denote the Parzen kernel function. This non-parametric approximation will converge to the true density $P\left(\vec{\mathbf{x}}_i|C_j\right)$ while L is close to infinity. However, it is also time consuming to compute the estimation result. A simplified version can be writen as following

$$P_{NN}\left(\vec{\mathbf{x}}_i|C_j\right) = \frac{1}{R}\sum_{r=1}^{R} K\left(\vec{\mathbf{x}}_i - NN_{C_j}^{r}(\vec{\mathbf{x}}_i)\right) \tag{10}$$

where $NN_{C_j}^{r}(\vec{\mathbf{x}}_i)$ denotes the r^{th} nearest neighbor of $\vec{\mathbf{x}}_i$ among all local features that belong to class C_j. R denotes the number of the nearest neighbors, and it can be 1 in an extreme case. In our experiments, we just use the Euclidean kernel function and set R to be 1. Now the above Eq.(10) is further simplified as

$$\mathbf{v_{ij}} = P\left(C_j|\vec{\mathbf{x}}_i\right) \propto \|\vec{\mathbf{x}}_i - NN_{C_j}(\vec{\mathbf{x}}_i)\|_2^2 \tag{11}$$

where $NN_{C_j}(\vec{\mathbf{x}}_i)$ denotes the nearest neighbor of $\vec{\mathbf{x}}_i$ in the sub feature space which contains all local features with class label C_j. Thus we get an extremely simple but efficient feature coding method for describing each local feature $\vec{\mathbf{x}}_i$ with the visual characteristics combined with class information.

To embed the locality and saliency into our feature coding method, we change the Eq.(11) to Eq.(12). When we compute the feature code $\vec{\mathbf{v}}_i$ for a local feature $\vec{\mathbf{x}}_i$, the $k+1$ nearest neighbors of $\vec{\mathbf{x}}_i$ in the whole local feature space of the training image set are searched in advance. The k nearest neighbors constitute the local feature subset S_b while the $(k+1)$-st nearest neighbor is used to measure the saliency. Thus, in the following algorithm, the local feature space for computing $\vec{\mathbf{v}}_i$ is shrinked to a small subset S_b. In addition, to obtain the saliency power, we subtract a distance $distB$ between $\vec{\mathbf{x}}_i$ and its $(k+1)$-st nearest neighbor as in [11].

$$\mathbf{v_{ij}} \propto \begin{cases} \|\vec{\mathbf{x}}_i - NN_{C_j}^{S_b}(\vec{\mathbf{x}}_i)\|_2^2 - distB & \textbf{if } C_j \in class(S_b) \\ 0 & \textbf{otherwise} \end{cases} \tag{12}$$

3.2 Discussion

In Fig. 1, we show an example about the difference of feature coding between traditional BOW based methods and our LDC. In the BOW based methods, each local feature $\vec{\mathbf{x}}_i$ is quantized to a feature vector $\vec{\mathbf{v}}_i$ based on visual words (clustered centers denoted as red points) as shown in the top part of Fig.1. For simplicity, only 4 cluster centers in the feature space and 3 classes are showed in the figure. In our LDC, as shown in the bottom part of Fig. 1, we do not use the visual words (green points) to quantize the local query SIFT feature. Instead, we focus on computing the distance between a local feature and its nearest feature (red point) in each class. In this way, we adopt the **Local Feature-to-class** distance for feature coding. Thus, the encoded feature vectors using our LDC method have discriminative or supervised information of the image class.

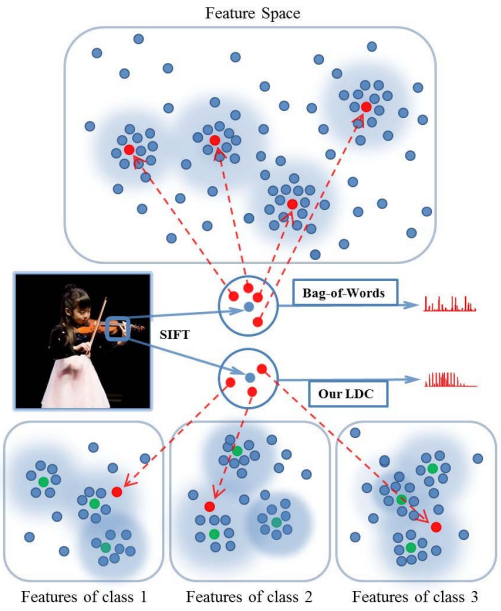

Figure 1: Illustration of the difference between Bag-of-Words based methods and our proposed LDC for feature coding.

4. EXPERIMENTAL RESULTS

In this section, we present extensive experimental results to validate the effectiveness of our LDC method for image classification.

4.1 Implementation Details

Our approach is tested on four popularly used benchmarks in the literature: Caltech101 [6], UIUC 8-Sports [9], 17-Class Flowers [12], and 7-Class PPMI [18]. For baseline methods, we use 6 state-of-the-art image classification methods denoted as: NBNN [2], NBNN Kernel [14], Local NBNN [11], ScSPM [8], LLC [16], and LSC [10]. We use the publicly available source codes or binaries provided by the authors with default parameters. For fair comparison, we adopt the same experimental settings for all methods, including image size, feature extraction strategy, and number of dictionary elements. In our experiments, we densely sample SIFT on the grey-scale image. A 3 levels SPM, such as 1×1, 2×2, 4×4 cells, is used to combine the spatial information. Finally, the linear SVM [5] is adopted for image classification.

4.2 Results and Analysis

From the Table 1, we can see that the improvments on Caltech101, UIUC 8-Sports, and 7-Class PPMI are 2%, 4% and 10%, respectively. On 17-Class Flowers, the improvement is relatively unapparent, it is probably due to the var-

Table 1: Accuracy on 4 data sets

Method	Caltech101	Event8	Flowers17	PPMI7
NBNN [2]	65.0±1.1	74.6±1.2	63.2±3.6	72.4
NBNN Kernel [14]	61.3±0.2	–	–	–
Local NBNN [11]	66.1±1.1	74.7±2.0	63.7±3.3	76.6
SPM [8]	62.5±0.9	80.0±1.7	67.2±3.8	53.3
LLC [16]	65.3±1.2	81.8±1.5	71.5±2.8	66.3
LSC [10]	68.6±0.7	82.8±2.0	73.5±3.1	71.6
Our	**70.9±0.5**	**86.5±1.1**	**73.6±2.2**	**83.3**

ious size of flowers contained in different images.

Caltech101: Caltech101 [6] contains 101 categories of object images. For convenience, we set the feature extraction parameters according to Local NBNN [2]. Each image is resized to be no bigger than 300 in height and width. 15 training images and 15 test images are randomly selected for each category. Multi-scale SIFT features(16×16, 24×24, 36×36) with step size 3 are extracted. The accuracy results are showed in the second column of Table 1. Results for all baseline methods except LLC are borrowed from Local NBNN [2] with the same carefully checked parameters, and the result for LLC [16] is obtained by running the code provided by the authors [16].

UIUC 8-Sports: This data set contains 8 sport categories [9]. Each image is resized to be no bigger than 400 in height and width. 70 training images and 60 test images are randomly selected for each category. Single scale SIFT features (16×16) with step size 4 are extracted. The accuracy results are showed in the third column of Table 1. Results for ScSPM [8], LLC [16] and LSC [10] are borrowed from LSC [10]. Results for NBNN [16] and Local NBNN [11] are obtained by running the code provided by the authors.

17-Class Flowers: This data set contains 17 categories of flower images [12]. Each image is resized to be no bigger than 400 in height and width. Three splits of training and test images provided in the data set are used for our experiment. Single scale SIFT features (16×16) with step size 4 are extracted. The accuracy results are showed in the fourth column of Table 1. The results for the baseline methods are obtained by running the codes provided by the authors. From the results, we can see that our method has the comparable result as LSC [10], and is much better than other methods with at least 2% improvement.

7-Class PPMI: This data set contains 7 instrument categories of images with a fixed size 258 [18]. Here we just use images that contain a person playing instrument(PPMI+). Single scale SIFT features (16×16) with step size 4 are extracted. Due to the single split of 100 training images and 100 test images provided by the author, we just give the average accuracy without standard deviation in the last column of Table 1. Based on the results, we can see that our method improves the image classification performance significantly.

5. CONCLUSIONS

In this paper, based on Local NBNN, by using the distance between a local feature and its nearest neighbor in each class, we introduce a discriminative feature coding method for image classification, which completely reserves the locality and saliency of Local NBNN. Then, SPM is used to construct final visual representation. Extensive experimental results on four well-known benchmarks demonstrate the effectiveness of our LDC feature coding method compared with 6 state-of-the-art BOW and NBNN based methods.

6. ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China (61272329, 61225009), and Beijing Natural Science Foundation (4131004), also by the Singapore National Research Foundation under International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

7. REFERENCES

[1] R. Behmo, P. Marcombes, A. Dalalyan, and V. Prinet. Towards optimal naive bayes nearest neighbor. In *ECCV*, 2010.

[2] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *CVPR*, 2008.

[3] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on ECCV*, 2004.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley Interscience, 2 edition, 2000.

[5] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*.

[6] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In *Workshop on CVPR*, 2004.

[7] Y. Huang, K. Huang, Y. Yu, and T. Tan. Salient coding for image classification. In *CVPR*, 2011.

[8] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[9] L.-J. Li and F.-F. Li. What, where and who? classifying events by scene and object recognition. In *ICCV*, 2007.

[10] L. Liu, L. Wang, and X. Liu. In defense of soft-assignment coding. In *ICCV*, 2011.

[11] S. McCann and D. G. Lowe. Local naive bayes nearest neighbor for image classification. In *CVPR*, 2012.

[12] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *CVPR*, 2006.

[13] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*.

[14] T. Tuytelaars, M. Fritz, K. Saenko, and T. Darrell. The nbnn kernel. In *ICCV*, 2011.

[15] J. van Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders. Kernel codebooks for scene categorization. In *ECCV (3)*, 2008.

[16] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010.

[17] J. Yang, K. Yu, Y. Gong, and T. S. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 2009.

[18] B. Yao and L. Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *CVPR*, San Francisco, USA, June 2010.