This CVPR2015 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Beyond Spatial Pooling: Fine-Grained Representation Learning in Multiple Domains

Chi Li Austin Reiter Gregory D. Hager
Department of Computer Science, Johns Hopkins University
chi_li@jhu.edu, {areiter, hager}@cs.jhu.edu

Abstract

Object recognition systems have shown great progress over recent years. However, creating object representations that are robust to changes in viewpoint while capturing local visual details continues to be a challenge. In particular, recent convolutional architectures employ spatial pooling to achieve scale and shift invariances, but they are still sensitive to out-of-plane rotations. In this paper, we formulate a probabilistic framework for analyzing the performance of pooling. This framework suggests two directions for improvement. First, we apply multiple scales of filters coupled with different pooling granularities, and second we make use of color as an additional pooling domain, thereby reducing the sensitivity to spatial deformations. We evaluate our algorithm on the object instance recognition task using two independent publicly available RGB-D datasets, and demonstrate significant improvements over the current state-of-the-art. In addition, we present a new dataset for industrial objects to further validate the effectiveness of our approach versus other state-of-the-art approaches for object recognition using RGB-D data.

1. Introduction

The core challenge of object recognition is to create representations that are robust to appearance variations. Recent advances in convolutional architectures [27, 26, 10, 8] have achieved success in learning object representations with minor scale and shift invariances. *Spatial Pooling*, which groups local features within spatial neighborhoods, is a key component to achieve those invariance properties.

The discrimination and invariance capabilities of the spatially pooled features can be examined with regard to the density of pooling regions which we refer to as *pooling granularity*. The Bag-of-words model, which can be viewed as the extreme case of coarse pooling granularity, can tolerate large variations of object appearances caused by out-of-plane rotations. However, it loses the discriminative power provided by the spatial layout of features [29]. Conversely, fine-grained spatial pooling, which uses small and dense pooling regions (i.e., receptive fields), encodes fine-grained visual cues but is sensitive to spatial rearrangements in different object poses. This is demonstrated on the left block of Fig. 1, where the same object parts are pooled into different bins under an out-of-plane rotation. One solution is to deploy 'deep' convolutional architectures [27, 9, 40, 37, 23] which hierarchically pool local responses to boost the discrimination capability of features in the coarse-grained pooling. However, local characteristics are often lost due to the hierarchical pooling. This may not be desirable for the object instance recognition (as opposed to category recognition) where an object should be recognized as exactly the same one that has previously been seen. Recently, [17] integrates part-based modeling [4] into a deep convolutional neural network [27] to create more spatially aligned representations. They achieve state-of-the-art performance in public fine-grained object recognition benchmarks. This implies that robust fine-grained cues can be

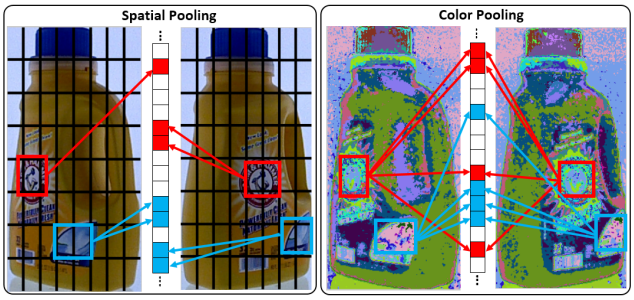

Figure 1. A comparison of fine-grained pooling between spatial (X, Y) and color (A, B) (last two channels in CIELAB) domains when an object undergoes a out-of-plane rotation. Fine-grained gridding (8×8) is performed in both domains. Pooling indices in the color domain are shown by different colors in the images. Pooling results for all pixels in two local patterns (enclosed by red and blue rectangles) are shown between pairs of images in each block. Correct feature alignments are made by the color domain, but fail in the spatial domain.

captured if visual features are better aligned with each other during fine-grained pooling.

In this paper, we analyze the performance of pooling-based convolutional architectures, and propose a simple but effective solution of pooling beyond spatial domain using adaptive scales of filters, to address the feature misalignment problem. Our major contributions are three-fold. First, we formulate a probabilistic framework to mathematically explain how the pooling granularity affects the learned representation in terms of the overall discrimination and invariance. We also argue that fine-grained pooling can be improved with small-scaled filters and invariant pooling domains that are insensitive to object transformations (one example is the color domain shown on the right block of Fig. 1). Second, based on these ideas, a novel multi-scale and multi-domain pooling algorithm is presented to learn fine-grained representations typical for large-scale object instance recognition task. Small to large scales of filters are coupled with fine to coarse pooling granularities in multiple domains respectively, in order to encode both the localized and global visual cues. Finally, we describe a new JHUIT-50 dataset including 50 industrial objects. A new experiment setting is designed to fully evaluate the invariance of the representation with respect to 3D transformations. The proposed method shows significant improvement over the current state-of-the-art on two public large-scale RGB-D datasets [28, 39] and the JHUIT-50 dataset.

The rest of the paper is organized as follows. Sec. 2 provides a background review of the invariant representation learning. Sec. 3 introduces a probabilistic framework for pooling which motivates our proposed method explained in Sec. 4. Experiments are presented in Sec. 5 and we conclude the paper in Sec. 6.

2. Related Work

Invariant representation learning has been studied in the past with empirical validations [30, 35, 22, 31] and theoretical analyses [3, 2]. Spatial pooling is found to be critical to gain the shift invariance in both feature coding pipelines [29, 34, 25, 11] and deep convolutional neural networks [27, 37, 23, 15]. Recently, an unsupervised feature learning theory [2] proposed an invariant signature by characterizing the distribution of template responses within certain transformation groups. This idea is shared in the design of the TIRBM [41], where minor 2D affine transformations are modeled during training. Similarly, data augmentation, a trick commonly used in deep CNNs [15, 27], is functionally equivalent to this strategy. However, in this category of work, only invariance to 2D affine transformations at most can be guaranteed for general object classes and only a subset of transformations can be modeled in practice.

Pooling in input feature space [12, 16, 42] can smooth the representation for better invariance, but this tends to lose discrimination capabilities. Thus, spatial layouts [12, 16] or supervised labels [20] are employed to create discriminative features. Additionally, learning optimal spatial pooling configurations in multiple pooling scales has been attempted by supervised [33, 24, 38] and unsupervised [46, 21] techniques as well as segmentation priors [14]. This series of work uses fixed filter scales in the spatial pooling domain while our method couples the adaptive filter scales with pooling granularities and deploys additional pooling domains to overcome feature misalignments.

Various rotationally invariant 3D feature descriptors [18, 19, 45, 1] were proposed for 3D object recognition, but these have been out-performed by multi-cue kernel descriptors [7, 8] and hierarchical convolutional architectures [9, 5, 40] in large-scale settings [28, 39]. The state-of-the-art method [9] mainly uses high-level features, coarse-grained spatial pooling, and contrast normalization to alleviate large intra-class variance caused by 3D rotations. However, spatial pooling still dominates the feature learning in those approaches, which makes learned representations only invariant to limited views of an object. In this study, we demonstrate that pooling simple local features in invariant domains can significantly boost the recognition performance for the object instance recognition.

3. A Framework for Analysis of Pooling

An overview of the general pooling process in a convolutional architecture is shown in Fig. 2. Filter responses associated with each pooling state are activated by feature filters convolved over visual signals. In the case of spatial pooling, pooling states are pixels in normalized image coordinates. A pooling operator extracts some statistics over filter responses within neighborhoods of pooling states. Few theoretical investigations have been presented in the literature to explain why pooling is critical in creating invariant representations. One pooling theory was proposed by Boureau [13] in the context of hard-assignment coding. It assumes that filter responses in a pooling region have identical and independent Bernoulli distributions given an object class. These conditions restrict the theory from generalizing to more complex scenarios. In this section, we develop a novel probabilistic view for pooling to resolve the aforementioned issues, which in turn motivates the proposed feature pooling algorithm in Sec. 4.

3.1. Interpretation of Invariance and Discrimination

Consider a pooling domain $S = \{s_1, \cdots, s_N\}$ where pooling state s_j with $1 \leq j \leq N$ is a coordinate over which pooling takes place. For example, in the case of RGB-D data, S can be a set of spatial coordinates or color values, corresponding to spatial and color domains.

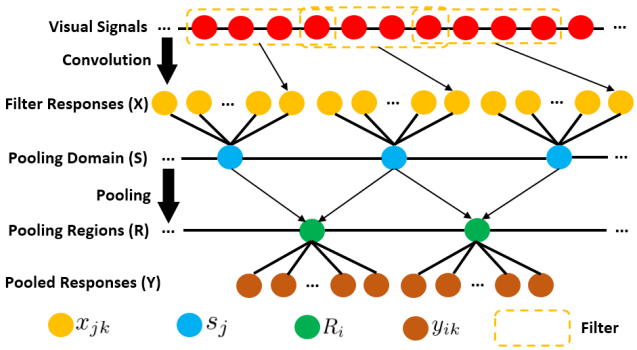

Figure 2. Demonstration of a general pooling process and related notations used in Sec. 3. At the top layer, we only show the convolution of one filter with one single scale. [Best viewed in color]

We now introduce a set of K filters $D = \{d_1, d_2, \cdots, d_K\}$. In the context of feature coding, these filters are codewords learned by dictionary learning techniques. Note that filters are not necessarily defined over the pooling domain (e.g., we could use the color domain to pool responses from spatial filters). Next, we define $X = (x_{11}, \cdots, x_{jk}, \cdots, x_{NK})$ as non-pooled representation for a data sample, where each $x_{jk} = (s_j, d_k)$ captures the activation strength of d_k at s_j (second row of Fig. 2). Each visual signal that occupies s_j contributes its K filter responses to the part of X associated with s_j. If two or more signals fall into the same s_j, we could compute the final response for each x_{jk} using any statistics (maximum value for example). Considering a random sampling of images generated by applying some transformation function $\mathcal{T}$ for object o_p, let $X^p = (x^p_{11}, \cdots, x^p_{jk}, \cdots, x^p_{NK})$ denotes the random vector of the filter responses with the distribution $P(X^p) = P(X|o_p)$. The $P(X^p)$ characterizes the distribution of the set of filter responses $G = \{X^p_i\}$ where X^p_i is a sample of X^p generated by $\mathcal{T}$.

We measure the variability of X^p with an invariance score J^p. Specifically, J^p is defined as the average Euclidean distance[1] between all samples in G:

$$\begin{aligned} J^p &= \frac{1}{t^2}\sum_{i=1}^{t}\sum_{j=1}^{t}\|X^p_i - X^p_j\|_2^2 = E(||X^p - \widetilde{X}^p||_2^2) \\ &= \sum_{j=1}^{N}\sum_{k=1}^{K} 2\text{Var}(x^p_{jk}) \end{aligned} \tag{1}$$

where $X^p_i, X^p_j \in G$. We use X^p and $\widetilde{X}^p$ as random variables for $\{X^p_i\}$ and $\{X^p_j\}$ respectively, which share the same distribution $P(X^p)$. As we can see, the invariance score J^p is actually the sum of variances of all dimensions in X^p. It measures how concentrated the representation is under the transformation $\mathcal{T}$. The smaller J^p, the better the stability of the descriptor.

Next, we formulate a distance metric $\mathcal{D}(X^p, X^q)$ between X^p and X^q given two object classes o_p and o_q as follows:

$$\mathcal{D}(X^p, X^q) = \frac{1}{2}\frac{\|\Delta E\|_2^2}{J^p + J^q} \tag{2}$$

where $\Delta E = E(X^p) - E(X^q)$. We could interpret the numerator and denominator in $\mathcal{D}(X^p, X^q)$ as the measurements of the discrimination and invariance properties of non-pooled representation X, respectively. In fact, $\mathcal{D}(X^p, X^q)$ can be derived as the lower bound of the Bhattacharyya distance metric $\mathcal{D}_B(X^p, X^q)$ given that $P(X^p)$ and $P(X^q)$ follow multivariate normal distributions with covariances Σ_p and Σ_q. This can be shown as follows:

$$\begin{aligned} \mathcal{D}_B(X^p, X^q) &= \frac{1}{8}\Delta E^\top \bar{\Sigma}^{-1}\Delta E + \frac{1}{2}\ln\frac{|\Sigma|}{\sqrt{|\Sigma_p||\Sigma_q|}} \\ &\geq \frac{1}{8}\Delta E^\top (U\bar{\Lambda}^{-1}U^\top)\Delta E \\ &\geq \frac{1}{8}\frac{\|U^\top \Delta E\|_2^2}{tr(\bar{\Lambda})} \\ &= \frac{1}{2}\frac{\|\Delta E\|_2^2}{J^p + J^q} \\ &= \mathcal{D}(X^p, X^q) \end{aligned} \tag{3}$$

where $\bar{\Sigma} = \frac{\Sigma_p + \Sigma_q}{2}$ with eigen-decomposition $\bar{\Sigma} = U\bar{\Lambda}U^\top$. The second step is obtained by the Cauchy-Schwarz inequality and the third step is derived according to the mediant inequality [2]. The final step follows by $\|Ux\| = \|x\|$ if U is unitary and the $tr(\bar{\Lambda}) = tr(\bar{\Sigma}) = \frac{1}{4}(J_p + J_q)$. **Note that random variables are allowed to be dependent on each other in this derivation.** From the perspective of the lower bound of $\mathcal{D}_B(X^p, X^q)$, $\mathcal{D}(X^p, X^q)$ characterizes the most ambiguous region between two feature distributions. Notice that $\mathcal{D}(X^p, X^q)$ shares a similar form with the objective in linear discriminant analysis (LDA) and distribution separability in [13] using a signal-to-noise ratio.

3.2. Variance Reduction via Pooling

In this section, we show that pooling filter responses within regions in S reduces the variance of the non-pooled representation X^p. Let $\mathcal{R} = \{R_1, \cdots, R_M\}$ be a partition of S (i.e., a set of non-overlapping pooling regions) and assume max pooling is used [3]. In turn, we define a new random variable $y_{ik} = \max_{s_j \in R_i} x_{jk}$ that represents the pooled filter response in pooling region R_i. Analogous to X^p, we then define the random vector $Y^p_{\mathcal{R}} =$

[1] This corresponds to the distance metric in linear SVM which is used in this study.

[2] $\frac{b}{a} + \frac{d}{c} \geq \frac{b+d}{a+c}$ if $a, b, c, d \geq 0$

[3] We choose max pooling operator [36] for our main analysis because many studies [13, 11] show its better performance over average pooling.

$(y^p_{11}, y^p_{12}, \cdots, y^p_{MK})$. $J^p_{\mathcal{R}}$ is the invariance score of the pooled representation $Y^p_{\mathcal{R}}$. We can then prove the following result using the fact that $\text{Var}(\max_i X_i) \leq \sum_i \text{Var}(X_i)$ [4]:

$$J^p_{\mathcal{R}} = \sum_{k=1}^{K}\sum_{i=1}^{M} 2\text{Var}(y^p_{ik}) \leq \sum_{k=1}^{K}\sum_{j=1}^{N} 2\text{Var}(x^p_{jk}) = J^p \quad (4)$$

In short, max pooled feature $Y^p_{\mathcal{R}}$ has lower variance than non-pooled feature X^p, which means $Y^p_{\mathcal{R}}$ is less sensitive to transformations $\mathcal{T}$ than X^p. The same can be shown for average pooling [5] because $\text{Var}(\frac{1}{N}\sum_i X_i) \leq \sum_i \text{Var}(X_i)$. Furthermore, J^p is a very loose upper bound for $J^p_{\mathcal{R}}$ in Eq. 4. The equality is achieved in the asymptotic regime when one random variable is always greater than remaining ones with zero variance. Therefore, $J^p_{\mathcal{R}}$ is much smaller than J^p in practice.

Furthermore, in the case of intersecting pooling regions $\widehat{\mathcal{R}} = \{\widehat{R}_1, \cdots, \widehat{R}_M\}$, we can find a non-overlapping set $\mathcal{R} = \{R_1, \cdots, R_M\}$ subject to $\cup R_i = \cup \widehat{R}_i$ and $R_i \subseteq \widehat{R}_i$. Then we can get $J^p_{\widehat{\mathcal{R}}} \leq J^p_{\mathcal{R}}$ because each $\widehat{R}_i$ further pools the result of R_i so that the invariance score decreases according to Eq. 4. Thus, the overlapping pooling scheme achieves even lower variance than the non-overlapping case, though it tends to decrease $\|\Delta E_{\widehat{\mathcal{R}}}\|_2^2 = \|E(Y^p_{\widehat{\mathcal{R}}}) - E(Y^q_{\widehat{\mathcal{R}}})\|_2^2$ since each pooling region is more likely to acquire high activation responses when it is enlarged. For simplicity, we continue to assume pooling regions are a partition in the following discussion.

Analogous to Eq. 3, we can also write the distance between $Y^p_{\mathcal{R}}$ and $Y^q_{\mathcal{R}}$ for object classes o_p and o_q as follows:

$$\mathcal{D}(Y^p_{\mathcal{R}}, Y^q_{\mathcal{R}}; \mathcal{R}) = \frac{1}{2}\frac{\|\Delta E_{\mathcal{R}}\|_2^2}{J^p_{\mathcal{R}} + J^q_{\mathcal{R}}} \quad (5)$$

where $\Delta E_{\mathcal{R}} = E(Y^p_{\mathcal{R}}) - E(Y^q_{\mathcal{R}})$. It is clear that greater discrimination $\|\Delta E_R\|_2^2$ and lower variance $J^p_{\mathcal{R}} + J^q_{\mathcal{R}}$ lead to better separability and in turn easier classification.

3.3. Conclusions and Discussions

The above probabilistic framework for pooling yields three major conclusions:

1. As pooling granularity changes from fine to coarse levels, pooled features have better invariance (smaller $J^p_{\mathcal{R}}$) but less discrimination (smaller $\|\Delta E_{\mathcal{R}}\|_2^2$).

2. Small-scale filters achieve better invariance than the large-scale ones in fine-grained pooling.

3. Pooling domains that are insensitive to transformations obtain better invariance in fine-grained pooling.

[4] This is proved by the Theorem 1 in the supplementary material

[5] It is equivalent to sum pooling in the context of Eq. 5

The first point follows from to Eq. 4. $J^p_{\mathcal{R}}$ is monotonically decreasing (i.e., invariance of $Y^p_{\mathcal{R}}$ is increasing) with growing size of pooling regions. This can be shown by replacing the left and right sides in Eq. 4 with variances of pooled features from small and large pooling regions, respectively. On the other hand, the discrimination term $\|\Delta E_{\mathcal{R}}\|_2^2 = \sum_{k=1}^{K}\sum_{j=1}^{M} |E(y^p_{jk}) - E(y^q_{jk})|^2$ tends to decrease due to smaller M at a coarse pooling granularity, especially when y_{jk} is bounded in most of the feature encoding algorithms. One good tradeoff between invariance $J^p_{\mathcal{R}} + J^q_{\mathcal{R}}$ and discrimination $\|\Delta E_{\mathcal{R}}\|_2^2$ to get large $\mathcal{D}(Y^p_{\mathcal{R}}, Y^q_{\mathcal{R}}; \mathcal{R})$ is made by 'deep' representations [27, 9, 5, 40, 26, 37, 23], which augments discrimination capabilities in coarse-grained pooling with highly class-specific filters. In this work, we pursue a good tradeoff along the other direction in which the feature invariance is enhanced in fine-grained pooling.

Next, we jointly analyze the last two points by looking more closely at $\text{Var}(x^p_{jk})$. In the context of fine-grained pooling where the number of pooling regions M is large, the invariance score $J^p_{\mathcal{R}}$ significantly drops if the variance of filter responses at each pooling state $\text{Var}(x^p_{jk})$ is reduced whereas the discrimination term $\|\Delta E_R\|_2^2$ is dominated by M and remains roughly the same. Therefore, we explore two ways to reduce $\text{Var}(x^p_{jk})$ for better separability $D(Y^p_{\mathcal{R}}, Y^q_{\mathcal{R}})$ in fine-grained pooling. Specifically, we observe that $P(x^p_{jk})$ can be decomposed into the following two forms:

$$P(x^p_{jk}) = P(d_k|s_j, o_p)P(s_j|o_p) \quad (6)$$

$$P(x^p_{jk}) = P(s_j|d_k, o_p)P(d_k|o_p) \quad (7)$$

As a result, $\text{Var}(x^p_{jk})$ is positively proportional to $\text{Var}(d_k|s_j, o_p)$ or $\text{Var}(s_j|d_k, o_p)$ [6]. Then we could make $\text{Var}(x^p_{jk})$ smaller by decreasing either $\text{Var}(d_k|s_j, o_p)$ or $\text{Var}(s_j|d_k, o_p)$. First, reducing $\text{Var}(d_k|s_j, o_p)$ can be interpreted as choosing filters that have smaller variance across the pooling domain S. Given a fixed filter learning method, smaller $\text{Var}(d_k|s_j, o_p)$ is achieved via small-scale filters rather than large-scale ones because the value changes of local regions are less than large areas in convolution. However, large-scale filters are prone to create better discrimination, which is more favored in coarse-grained pooling. Second, reducing $\text{Var}(s_j|d_k, o_p)$ is equivalent to constructing a pooling domain where appearance features have better alignments at each s_j. In other words, a more robust pooling domain with respect to transformations leads to smaller variance of filter responses at each pooling state s_j. Considering 3D transformations, spatial layouts of the transformed object samples change sharply while color configurations

[6] This is proven by Theorem 3 in the supplementary material

are typically aligned across different poses [7]. The possible color misalignment is caused by different lighting conditions, which can be largely alleviated by a good choice of color space and the pooling process. This fact motivates us to exploit the color domain as an example of an invariant domain in this study .

Although the spirit of discrimination-invariance tradeoff is already revealed by some kernel learning techniques [44], our framework associates it with pooling operator in the context of the convolutional architecture. As far as we know, we are the first to present this novel view and explore the way to make a good tradeoff. All the three conclusions derived in this section are empirically validated in Sec. 5.1.

4. Multi-Scale and Multi-Domain Pooling

The three theoretical views shown in Sec. 3.3 directly lead to the design of the multi-scale and multi-domain pooling algorithm presented in this section. Prior to going into the details of the proposed method, we first briefly explain the local feature we use. Specifically, we choose the rotationally invariant 3D descriptor CSHOT [19] as the raw features associated with each RGB-D image pixel. We modify the original CSHOT descriptor by decoupling the color and depth components. Then, dictionaries for each component are learned via hierarchical K-means and in turn feature codes are generated by a soft-assignment encoder [43, 32], which has been shown to perform as well as the sparse coding, but with much less computation. Soft-assignment coding can be formulated as follows:

$$\mu_j = \frac{\exp\left(\beta \hat{d}(x, d_j)\right)}{\sum_{k=1}^{n} \exp\left(\beta \hat{d}(x, d_k)\right)} \qquad (8)$$
$$s.t.\ \hat{d}(x, d_k) = \begin{cases} d(x, d_k) & : d_k \in \mathcal{N}_k(x) \\ +\infty & : d_k \notin \mathcal{N}_k(x) \end{cases}$$

where $\hat{d}(x, d_k)$ is the localized form of the original squared Euclidean distance $d(x, d_k)$ between raw visual signal x and codeword d_k and $\mathcal{N}_k(x)$ denotes the k-nearest neighbors of x defined by $d(x, d_k)$ within dictionary $D = \{d_1, d_2, \cdots, d_K\}$ (i.e., filters defined in Sec. 3.1). β is a smoothing parameter with negative value. Depth and color feature codes are concatenated as filter responses for x. We keep the feature extraction simple in order to isolate the contributions in our proposed pooling algorithm.

Next, we use the conclusions of Sec. 3.3 to guide the design of our feature learning algorithm. Unlike spatial pyramid pooling, where filter responses with fixed scale go into different pooling levels, the second point in Sec. 3.3 inspires us to pool responses from small-scale filters in fine-grained levels while large-scale filter responses are pooled in coarse-grained levels. In our implementation, we adopt the max pooling operator and adjust the scales of filters (i.e., codewords) by altering the 3D radius of CSHOT feature. Filters at each scale are learned independently via hierarchical K-means. Moreover, we employ the color domain for feature pooling in addition to the spatial domain (the third point in Sec. 3.3). Therefore, each CSHOT filter response goes into a pooling region based on the color value of the RGB-D image pixel associated with it and the max pooling is applied for all responses within the same pooling region. Note that spatial domain is not abandoned because spatially aligned features under slight change of view points could still benefit the recognition (shown in Sec. 5.2).

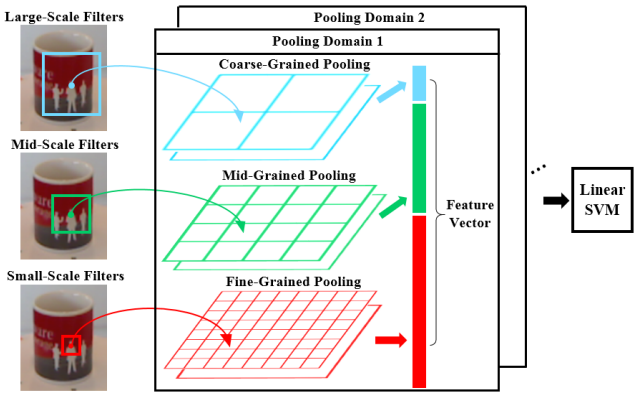

Figure 3. Overview of multi-scale and multi-domain pooling architecture.

In summary, the proposed method (shown in Fig. 3) is evolved from the common coding-pooling pipeline [29, 11, 10], but it conducts an adaptive pooling scheme on convolutional filter responses in multiple scales and both the color as well as spatial domains. Pooled features from fine to coarse pooling levels across different domains are concatenated together to generate the final representation and a linear SVM is used for the classification.

5. Experiments

We perform experiments on three RGB-D datasets: UW-RGBD[28], BigBIRD[39] and our own JHUIT-50 dataset. CSHOT features [19] are extracted densely over each point in the point cloud that is generated from color and depth images. We alter the radius of the CSHOT feature to adjust the scale of the filters. Depth and color components in the raw CSHOT feature are decoupled into two feature vectors. Dictionaries with 200 codewords are learned by hierarchical K-means for each component. Note that the dictionary size is fixed across CSHOT filters with different radii. Finally, a soft-assignment encoder [43, 32] is used to generate feature codes of both components which are further concatenated as the local feature code. We choose the number of near-

[7] Photometric variation of object appearances are much smoother in general.

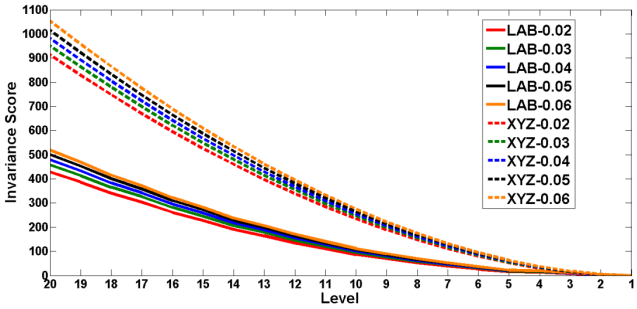

Figure 4. Comparison of the variances in different filter scales, pooling granularities and domains. The legend name 'domain-radius' indicates the pooling domain and the radius of CSHOT features respectively. [best viewed in color]

est neighbors K as $K = 45$ and the smoothing factor β as $\beta = -4.0$ in soft encoding (Eq. 8). All parameters are selected by cross-validation on a subset of the UW-RGBD dataset [8]. Feature codes within the same pooling region are further normalized using the L2-norm. We choose the CIELAB color space as the color pooling domain since we found that it achieves better performance than both RGB and HSV color spaces. The spatial domain is constructed in 3D space (XYZ). Each channel in the spatial and color domains is normalized to $[0, 1]$ to gain scale invariance. The feature codes are pooled inside the cells of the pyramid with multiple levels. Each level is constructed with a different granularity by gridding in a particular domain. Specifically, level-k in either the spatial (XYZ) or color (LAB) domain is constructed by $k \times k \times k$ grids. Pooled features across different levels and domains are concatenated as the final representation.

5.1. Variance Reduction via Pooling

We first conduct an experiment to verify the three conclusions derived from the probabilistic framework in Sec 3.3. The experimental results obtained in this section are commonly observed in almost all objects in those three datasets. For simplicity, we choose the object 'mixed_berry'[9] from BigBIRD as the representative for analysis. The variance in object representation is rooted from different object poses under 3D transformations. A detailed description about the object data can be found in Sec 5.3. CSHOT features with radii ranging from 0.02m to 0.06m are extracted and pooled from level-1 to level-20 separately in both the XYZ and LAB domains. Fig. 4 shows the empirical invariance scores of Eq. 1 across different levels and domains. Three major observations follow: (1) The invariance of the representation generated by all scales of filters in either domain increases via pooling [10] and maximal invariance is achieved by pooling in the entire domain (i.e., bag-of-words model). (2) Large-scale filters retain greater variance in all levels and both domains than small-scale filters. (3) The color domain exhibits much less variance in the learned representation than the spatial domain in all pooling granularities. These three observations empirically verify the three major points concluded in Sec 3.3. This further supports the proposed algorithm in Sec. 4.

[8] First 30 object instances.

[9] It is short for 'eating_right_for_healthy_living_mixed_berry'.

[10] Smaller invariance score indicates better invariance.

Algorithm	Acc.	Algorithm	Acc.
Linear SVM [28]	73.9	**XYZ-S-5**	85.5
NonLinear SVM [28]	74.8	**LAB-S-5**	89.8
RF [28]	73.1	**All-S-5**	93.3
CKM Desc. [5]	90.8	**XYZ-M-5**	87.9
Kernel Desc. [6]	91.2	**LAB-M-5**	91.9
HMP-All [8]	92.8	**All-M-5**	94.1

Table 1. Testing accuracies (%) of different methods on UW-RGBD. Variants of proposed method are marked in bold type.

5.2. UW-RGBD Object Dataset

Next, we evaluate our method on the UW-RGBD dataset which contains 300 daily object instances taken from different view points. The objects in this dataset are segmented from the background using color and depth cues. Both textured and textureless objects in various poses make this dataset challenging for recognition. In this study, the proposed fine-grained representation is tested on the object instance recognition task with the leave-sequence-out setting. Table 1 reports the testing accuracies of the proposed methods and comparative algorithms in the literature. The algorithm name for different variants of the proposed method (marked in bold type in Table 1) is formatted as 'domain-type-level'. More specifically, 'domain' indicates the pooling domain from LAB, XYZ or both, 'type' includes 'S' and 'M' referring to single and multiple scales of filters, and 'level' specifies the number of stacked levels used in the pyramid from level-1. For type 'S', we use the CSHOT feature with radius 0.03 across all experiments. In type 'M', feature responses from five scales of CSHOT filters from 0.02m to 0.06m with interval 0.01m are pooled within levels from 5 to 1 respectively. Table 1 shows that the multi-scale and multi-domain pooling scheme ('All-M-5') achieves the best result at 94.1%, which outperforms the current state-of-the-art with 92.8%. It also shows that the XYZ domain performs worse than the LAB domain and the combination of both domains achieves the best performance. This is because the view point changes in this experiment design ($15 \sim 20$ degrees) do not significantly disrupt the spatial layout for some typical objects with nearly homogeneous appearances, like a ball. Thus, correct feature alignments can be captured by spatial pooling to benefit the overall recognition. Lastly, multiple-scale filter (M) is consistently superior to single single-scale filter (S) in terms of

Algorithm	Acc.	Algorithm	Acc.
XYZ-S-1	75.1	LAB-S-1	75.1
XYZ-S-2	84.3	LAB-S-2	87.8
XYZ-S-3	86.1	LAB-S-3	88.3
XYZ-S-4	85.7	LAB-S-4	89.2
XYZ-S-5	85.5	LAB-S-5	89.6

Table 2. Testing accuracies (%) of different number of stacked levels in spatial (XYZ) and color (LAB) domains.

Algorithm	Acc.	Algorithm	Acc.
OUR-CVFH [1]	10.2	**XYZ-S-8**	31.2
ESF [45]	23.1	**LAB-S-8**	85.9
Kernel Descr. [6]	85.5	**ALL-S-8**	82.5
HMP-Depth [8]	35.1	**XYZ-M-8**	36.4
HMP-Color [8]	84.4	**LAB-M-8**	88.4
HMP-All [8]	80.8	**All-M-8**	84.6

Table 3. Testing accuracies (%) of different methods on BigBIRD. Variants of proposed method are marked in bold type.

the recognition rate.

Another experiment was performed to analyze how pooling granularity affects classification. Only single-scale filters are used in order to eliminate the effect of multi-scale filters. Table 2 reports the accuracies achieved by different numbers of stacked levels in XYZ and LAB. Accuracies in level-1 are the same between XYZ and LAB because the bag-of-words modeling results in the same pooled features regardless of the domain. Beyond level-1, the color domain consistently achieves higher accuracies than the spatial domain. Also, when pooling is performed over fine-grained levels, color pooling is able to continuously boost the recognition rates while spatial pooling fails to do so. This observation substantiates that the better invariance achieved by the color domain (shown in Fig. 4) helps to utilize the discrimination power in fine-grained levels.

5.3. BigBIRD Object Dataset

We also tested our algorithm on the BigBIRD dataset [39]. This dataset contains 125 daily objects in which many object instances are very similar to each other. Each object has 600 Kinect-style RGB-D images covering five fixed viewing angles from 0 to 90 degrees [11]. As a result, the pose variation in BigBIRD is much larger than UW-RGBD in which object data is captured in three viewing angles of 30, 45 and 60 degrees. In turn, we adopt an architecture with a maximum of 8 stacked levels in both domains, in order to further analyze fine-grained pooling under a larger subset of 3D transformations. As far as we know, there is no evaluation metric for the object instance recognition on BigBIRD. Thus, we follow the similar experiment design in UW-RGBD to use sequences of the first, third and fifth viewing angles defined in BigBIRD for training and the remaining two for testing. We choose the state-of-the-art HMP[8], kernel descriptor [6] on UW-RGBD dataset and two rotationally invariant 3D descriptors OUR-CVFH [1] and ESF [45] for comparison. Those methods are implemented with source codes provided by the authors[12] and the PCL library[13]. Parameters for all comparative methods are optimized by cross-validation on the first 30 objects. From Table 3, we observe our proposed architecture 'LAB-M-8' [14] achieves the highest recognition rate. Unlike the results in UW-RGBD, the combined domain is inferior to the color domain only. This is mainly because spatial pooling performs much worse than color pooling in both single and multiple filter scales.

[11] Though this dataset provides high-resolution color images and full 3D meshes, we only use the RGB-D images in this study.

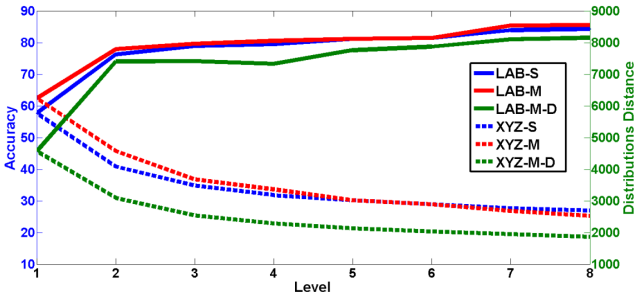

Figure 5. Classification accuracies at each level in pyramid and average distances (Eq. 5) between all object classes in color and spatial pooling domains.

For a more detailed analysis, we plot the recognition accuracies of each level (no stacking) in the color and spatial domains in Fig. 5. Clearly, the testing accuracies achieved by the spatial domain drop dramatically in fine-grained levels while the color domain continuously boosts the accuracies. Also, the multi-scale filters still perform better than the single-scale ones, which coincides with the observation in UW-RGBD. Finally, we calculate the average probabilistic distances of Eq. 5 between all pairs of object classes for pooled features using multi-scale filters. The solid (LAB-M-D) and dashed (XYZ-M-D) green lines show the average distances at each level in the LAB and XYZ domains, respectively. We can see that the distance metric derived in Eq. 5 is able to describe the general trend of the recognition performance, which further validates the probabilistic framework in Sec. 3.

5.4. JHUIT-50 Dataset

We present the JHUIT-50 dataset with a RGB-D camera [15] that contains 50 industrial objects and hand tools frequently used in mechanical operations. We segment each

[12] http://rgbd-dataset.cs.washington.edu/software.html

[13] http://pointclouds.org/

[14] Multiple scales of filters are specifically 0.02, 0.02, 0.02, 0.03, 0.03, 0.04, 0.04, 0.05, 0.05, 0.06 for levels from 8 to 1.

[15] PrimeSense Carmine 1.08 depth sensor is used.

Figure 6. Object examples in JHUIT-50 dataset. Left and right columns show two pairs of ambiguous object instances.

object from the background following the same procedures in the BigBIRD dataset. Fine-grained visual cues are often employed to distinguish these types of objects. For example, the left column of Fig. 6 shows two screwdrivers with only slight differences of texture patterns. Also, we treat different articulations of objects as separate object instances during recognition. The right column of Fig. 6 shows two configurations of a green clamp. We refer readers to the supplementary material for more details of this new dataset.

In the previous two experiments, testing data comes from sequences with fixed viewing angles. This constrained set of partial views may bias the evaluation of generalization performance towards a limited space in the entire viewing sphere, which is not desirable as a test for a realistic recognition scenario. In order to compensate for this drawback, we adopt two distinct collection procedures for training and testing data. On the training side, each object is placed on a turntable in increments of 7.2 degrees at three fixed camera viewing angles with 30, 45 and 60 degrees. This amounts to $\frac{360}{7.2} \times 3 = 150$ object views in total for training. For testing data, we manually move the camera around objects to sample another 150 random views of the object from the whole viewing sphere as the testing data. In this newly designed experiment setting, the testing data sampled from the full pose space contains larger pose variations than the previous two datasets. We deploy the same architecture with an 8 level pyramid used in BigBIRD on this dataset and the testing accuracies are reported in Table 4. We can clearly see that the experiment results on this dataset are similar to the previous two. First, color pooling and multi-scale filters are consistently superior to the spatial pooling and single-scale filters. Additionally, 'All-M-8' achieves the best result which significantly outperforms any others. Notice that spatial domain performs relatively better compared with the experiments on the BigBIRD dataset, though the pose variation is larger. This is mainly because the random testing views have overlaps with training views so that the spatial domain can positively contribute correct feature alignments for a subset of data.

Algorithm	Acc.	Algorithm	Acc.
OUR-CVFH [1]	45,1	**XYZ-S-8**	75.5
ESF [45]	76.8	**LAB-S-8**	88.6
Kernel Descr. [6]	82.1	**ALL-S-8**	90.5
HMP-Depth [8]	41.1	**XYZ-M-8**	76.6
HMP-Color [8]	81.4	**LAB-M-8**	90.1
HMP-All [8]	74.6	**All-M-8**	91.2

Table 4. Testing accuracies (%) of different methods on IT.

5.5. Limitations

Although the proposed method achieves improvement over the current state-of-the-art on the aforementioned three datasets, two major limitations remain. First, fine-grained pooling in high levels (> 8) results in feature vectors with more than one million dimensions though it is sparse due to the soft-assignment encoder. This prevents more fine-grained implementations on large-scale data. We could resolve this issue by using receptive field learning techniques [33, 24] to select a subset of pooling regions. Second, the color domain fails to generalize object poses across different object instances that have different color distributions, which makes it less applicable in object category recognition. Recall that any feature space could be used as a pooling domain in Sec. 3. A promising solution is constructing other invariant domains that capture the invariant properties for both object poses and category characteristics.

6. Conclusion and Future Work

In this paper, we have presented a fine-grained feature learning framework that is insensitive to common 3D transformations using multi-scale and multi-domain pooling. The three main conclusions of this work are that: (1) a good fine-grained representation can be learned by fine-grained pooling within domains that are insensitive to object transformations; (2) filter responses over small-scale areas are preferred in fine-grained pooling; (3) the spatial domain is much less favorable than color domains towards learning representations that are invariant to 3D transformations, typically in the case of fine-grained pooling. We demonstrated that the proposed feature learning architecture significantly outperforms the current state-of-the-art on both public and self-collected datasets.

We believe the theoretical pooling framework in this work can inspire a new design of feature learning architectures. For future work, not only can we explore new pooling domains with better invariance properties, but also new deep representations constructed beyond the spatial domain.

Acknowledgement

This work is supported by the National Science Foundation under Grant No. NRI-1227277.

References

[1] A. Aldoma, F. Tombari, R. B. Rusu, and M. Vincze. *OUR-CVFH–Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation*. 2012.

[2] F. Anselmi, J. Z. Leibo, L. Rosasco, J. Mutch, A. Tacchetti, and T. Poggio. Unsupervised learning of invariant representations in hierarchical architectures. *arXiv preprint arXiv:1311.4158*, 2013.

[3] Y. Bengio and Y. LeCun. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 2007.

[4] T. Berg and P. N. Belhumeur. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *CVPR*. IEEE, 2013.

[5] M. Blum, J. T. Springenberg, J. Wulfing, and M. Riedmiller. A learned feature descriptor for object recognition in rgb-d data. In *ICRA*. IEEE, 2012.

[6] L. Bo, K. Lai, X. Ren, and D. Fox. Object recognition with hierarchical kernel descriptors. In *CVPR*. IEEE, 2011.

[7] L. Bo, X. Ren, and D. Fox. Kernel descriptors for visual recognition. In *NIPS*, 2010.

[8] L. Bo, X. Ren, and D. Fox. Hierarchical matching pursuit for image classification: Architecture and fast algorithms. In *NIPS*, 2011.

[9] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. *ISER*, 2012.

[10] L. Bo, X. Ren, and D. Fox. Multipath sparse coding using hierarchical matching pursuit. In *CVPR*. IEEE, 2013.

[11] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*. IEEE, 2010.

[12] Y.-L. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals: multi-way local pooling for image recognition. In *ICCV*. IEEE, 2011.

[13] Y.-L. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *ICML*, 2010.

[14] L. Cao, R. Ji, Y. Gao, Y. Yang, and Q. Tian. Weakly supervised sparse coding with geometric consistency pooling. In *CVPR*. IEEE, 2012.

[15] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[16] A. Coates and A. Y. Ng. Selecting receptive fields in deep networks. In *NIPS*, 2011.

[17] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *ICML*, 2014.

[18] S. S. F. Tombari and L. D. Stefano. Unique signatures of histograms for local surface description. In *ECCV*, 2010.

[19] S. S. F. Tombari and L. D. Stefano. A combined texture-shape descriptor for enhanced 3d feature matching. *ICIP*, 2011.

[20] S. R. Fanello, N. Noceti, C. Ciliberto, G. Metta, and F. Odone. Ask the image: supervised pooling to preserve feature locality. In *CVPR*. IEEE, 2014.

[21] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV*. Springer, 2014.

[22] I. Goodfellow, H. Lee, Q. V. Le, A. Saxe, and A. Y. Ng. Measuring invariances in deep networks. In *NIPS*, 2009.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014.

[24] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *CVPR*. IEEE, 2012.

[25] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning invariant features through topographic filter maps. In *CVPR*. IEEE, 2009.

[26] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. L. Cun. Learning convolutional feature hierarchies for visual recognition. In *NIPS*, 2010.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[28] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *ICRA*, 2011.

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

[30] Y. LeCun, F. J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*. IEEE, 2004.

[31] Q. Liao, J. Z. Leibo, and T. Poggio. Learning invariant representations and applications to face verification. In *NIPS*, 2013.

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

[33] M. Malinowski and M. Fritz. Learnable pooling regions for image classification. *arXiv preprint arXiv:1301.3516*, 2013.

[34] J. Mutch and D. G. Lowe. Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision (IJCV)*, 2008.

[35] N. Pinto, Y. Barhomi, D. D. Cox, and J. J. DiCarlo. Comparing state-of-the-art visual features on invariant object recognition tasks. In *Applications of computer vision (WACV), 2011 IEEE workshop on*. IEEE, 2011.

[36] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 1999.

[37] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[38] K. Simonyan, A. Vedaldi, and A. Zisserman. Descriptor learning using convex optimisation. In *ECCV*. Springer, 2012.

[39] A. Singh, J. Sha, K. S. Narayan, T. Achim, and P. Abbeel. Bigbird: A large-scale 3d database of object instances. In *ICRA*, 2014.

[40] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Ng. Convolutional-recursive deep learning for 3d object classification. In *NIPS*, 2012.

[41] K. Sohn and H. Lee. Learning invariant representations with local transformations. *ICML*, 2012.

[42] S. Sukhbaatar, T. Makino, and K. Aihara. Auto-pooling: Learning to improve invariance of image features from image sequences. *arXiv preprint arXiv:1301.3323*, 2013.

[43] J. C. Van Gemert, C. J. Veenman, A. W. Smeulders, and J.-M. Geusebroek. Visual word ambiguity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2010.

[44] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *ICCV*. IEEE, 2007.

[45] W. Wohlkinger and M. Vincze. Ensemble of shape functions for 3d object classification. In *Robotics and Biomimetics (ROBIO)*. IEEE, 2011.

[46] C. Xu and N. Vasconcelos. Learning receptive fields for pooling from tensors of feature response. In *CVPR*. IEEE, 2014.