Efficient Maximum Appearance Search for Large-Scale Object Detection

Qiang Chen[1], Zheng Song[1], Rogerio Feris[2], Ankur Datta[2], Liangliang Cao[2],
Zhongyang Huang[3], Shuicheng Yan[1]

[1] National University of Singapore, Singapore, [2] IBM T.J. Watson Research Center

[3] Panasonic Singapore Laboratories, Singapore

{chenqiang,zheng.s,eleyans}@nus.edu.sg,

{rsferis,ankurd,liangliang.cao }@us.ibm.com, {zhongyang.huang}@sg.panasonic.com

Abstract

In recent years, efficiency of large-scale object detection has arisen as an important topic due to the exponential growth in the size of benchmark object detection datasets. Most current object detection methods focus on improving accuracy of large-scale object detection with efficiency being an afterthought. In this paper, we present the Efficient Maximum Appearance Search (EMAS) model which is an order of magnitude faster than the existing state-of-the-art large-scale object detection approaches, while maintaining comparable accuracy.

Our EMAS model consists of representing an image as an ensemble of densely sampled feature points with the proposed Pointwise Fisher Vector encoding method, so that the learnt discriminative scoring function can be applied locally. Consequently, the object detection problem is transformed into searching an image sub-area for maximum local appearance probability, thereby making EMAS an order of magnitude faster than the traditional detection methods. In addition, the proposed model is also suitable for incorporating global context at a negligible extra computational cost. EMAS can also incorporate fusion of multiple features, which greatly improves its performance in detecting multiple object categories. Our experiments show that the proposed algorithm can perform detection of 1000 object classes in less than one minute per image on the Image Net ILSVRC2012 dataset and for 107 object classes in less than 5 seconds per image for the SUN09 dataset using a single CPU.

1. Introduction

Large-scale object detection is an important vision problem concerned with detecting a large number of object categories and localizing them in a large number of images. Tremendous amount of recent research [1, 2, 3, 4, 5, 16] has focused on developing novel feature representations and classification algorithms to boost accuracy of large-scale object detection. A common thread that ties most of these state-of-the-art approaches together would be detection models that are designed to discriminate object shape from background on densely sampled sub-windows of images. Among these approaches, the template-based approaches, such as the popular Deformable Part Model (DPM) [2], use linear models constructed from a number of part templates of image gradient features. Since templates are sensitive to sampling scale and the pose of objects, inference of such models often entails exhaustively searching for the best template configuration regarding pose, scale, rotation, etc. Refinements to remedy this sampling problem brings extra computational cost, e.g DPM needs search the template configuration for best part combinations. Most object detection systems based on the aforementioned methods work at several seconds to tens of seconds per object

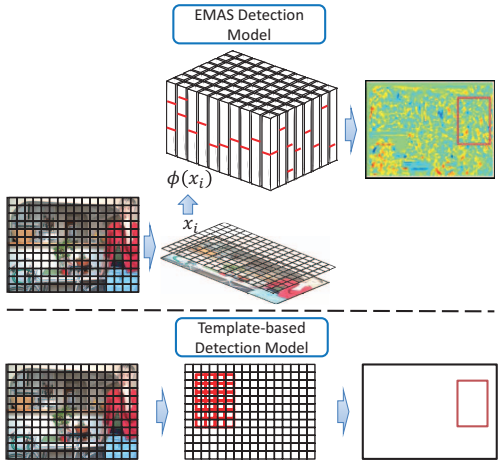


Figure 1: Upper part: the proposed EMAS model. The local feature x_i is first mapped to a high dimensional sparse vector $\phi(x_i)$ then the detection model can be applied locally to get the local confidence map. The model inference is achieved by an efficient maximum subarray search. Lower part: the template-based detection with exhaustive convolution over scales and positions.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.410

model per image [3, 4], and hence present significant barriers towards building systems that operate on a large number of images for a large number of object categories, i.e. towards practical large-scale object detection.

In this paper, we propose an Efficient Maximum Appearance Search (EMAS) model which is an order of magnitude faster as compared to the existing state-of-the-art approaches while maintaining state-of-the-art accuracy for large-scale object detection. Unlike the template-based approaches in which the learned model has to be applied to each testing window exhaustively, the key insight of our approach, as illustrated in Fig. 1, is that we represent an image as an ensemble of densely sampled feature points with the proposed Pointwise Fisher Vector encoding. This enriched local representation enables us to transform the object detection problem into searching for an image sub-window with maximum sum of object possibility, which can be performed extremely efficiently. The advantage of low computation complexity enables us to explore the large scale object detection problem with huge number of categories. We also show in our experiments that our appearance-based approach shows better results than the traditional shape-based approach when dealing with categories with large variance. Our contributions are thus as follows:

- We propose an efficient maximum appearance search model for large scale object detection. Our proposed EMAS applies the model locally to each transformed local points and the inference problem is transferred to searching the sub-window with maximum sum. As far as we know, this is the first model specifically designed for object detection with large number of categories, which makes it different from other works that focus on improving DPM model for efficiency [4, 31, 29].
- We propose the Pointwise Fisher Vector coding as the enriched local representation of our detection model. Our representation is motivated by the recent success of image classification work using Fisher Vector [10]. However, traditional Fisher Vector presentation requires a pooling and normalization operation in the image level, which makes it difficult to be used by sliding window search. In this work, we propose to maintain a local feature coding which benefits the discriminative power of the local patches. The key insight is that we extend Fisher Vector encoding to the point-level, which makes EMAS extremely efficient by enabling rapid maximum sub-window search. Moreover, this representation is able to construct a global form for multi-class detection and thus has the potential to search objects very efficiently in a large scale setting.
- We show state-of-the-art performance on two challenging datasets with large number of categories, i.e. SUN09 [20] and ILSVRC2012 [28]. Experimental evaluations show that the algorithm can perform detection of 1000 object classes in less than one minute per image on the Image Net ILSVRC2012 datasets and 107 object classes in about 5 seconds per image on the SUN09 dataset using single CPU with comparable performance to state-of-the-art algorithms.

2. Related Works

2.1. General Object Detection

Shape-based object detection models rely on discriminative shape templates using histograms of oriented gradients. Initially, Dalal and Triggs [1] used a single rigid template to build a detection model for pedestrians. Thereafter, the PASCAL VOC dataset [11] was released, comprising of objects with more deformable shapes like animals and vehicles. Hence, the single template model was extended to part-based models [2] by Felzenswalb et al., with inspiration from [26], to handle shape deformations. Although the deep convolution network [24] shows promising result on ImageNet, the part-based methods [12, 13, 23] are still the best-performing ones on the VOC dataset.

Previous research [16, 5, 7, 8, 9] has also explored BoW model detection. The MKL object detection [16] which uses kernel-based models and spatial pyramid (SP) feature combination achieves promising results but the computation cost is very high. Efficient Subwindow Search (ESS) [5, 7, 8, 9] tries to speed up the VQ-based BoW model using a branch and bound technique but often with much poorer performance on standard datasets. The main disadvantage of VQ is that it encodes the local feature as one specific visual word index, thus no complex local discriminative model can be build upon this.

The BoW-based model has the advantage of efficiency if one linear model can be applied and the possible theoretical computation cost is much less than the template-based approach. Suppose we use the same low level feature for both models, e.g. HOG. For a template model with $m \times n$ cells, we need to compute $m \times n$ times convolution at each pixel for each category test searching over the image. The search complexity is $\mathcal{O}(mnP)$ where P is the searching space complexity for an image. For a BoW model, the cost is separated into two parts, i.e. the local feature coding step and inference (dot-product) over the linear model. The cost of local feature coding step often increases with the codebook size K which is independent for each categorie. For multi-class object detection, the only cost addition is the inference cost which depends on the sparseness $\mathcal{E}$ of the coding. The sparseness is $1/K$ for hard Vector Quantization (VQ), and is around 3% for Fisher Vector coding (FV) [10] in our experiments. So, the inference complexity is $\mathcal{O}(\mathcal{E}P)$, which is much less than the template-based approach ($mn \gg \mathcal{E}$).

2.2. Feature Encoding

Recent feature encoding approaches, such as Sparse Coding [14] and Locality-constrained Linear Coding(LLC) [15], introduce soft assignment for local feature

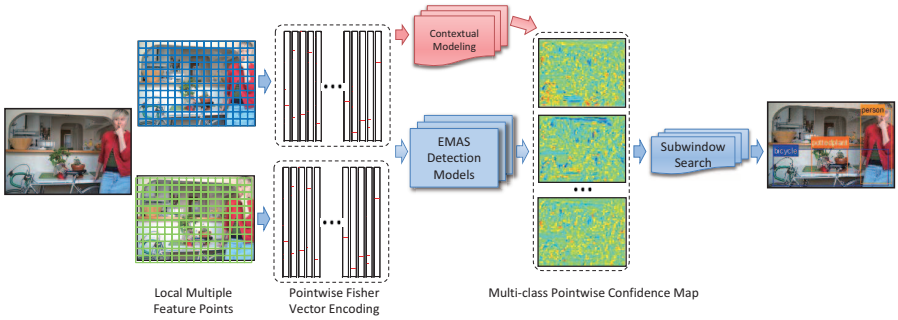


Figure 2: Framework illustration of Efficient Maximum Appearance Search.

quantization thereby improving previous discrete quantization methods and can been seen as an extension of Vector Quantization. For the recognition problem, these two coding methods benefit from large size codebooks as demonstrated in a recent survey [17]. The large codebook size and the introduction of soft assignment reduce the quantization error at the expense of increased computational cost. Recently, aggregation coding, such as, the Fisher Vector coding or the Super Vector coding, have demonstrated increased discriminative power of local features [17]. Fisher encoding [10] captures the average first and second order differences between local features and the centers of a Mixture of Gaussian Distributions learnt from general datasets, while the Super vector encoding [18] only focuses on the first order difference. Recently, G. Csurka et.al [25] extended the Fisher Vector coding to the patch level for the semantic segmentation task. As an extension of the previous approach, we propose to further extend the Fisher encoding to the point level. In other words, the scoring function operates on the point level as opposed to operating on the patch level as in [25].

2.3. Efficient Object Detection

In the past few years, various ways to reduce detection time have been explored in the literature. The cascade part-based model [4] accelerates the part-based models [2] by learning stagewise thresholds to fast reject negative sampling windows. Other methods improve the efficiency of current DPM model [31, 29]. The jumping windows method [3] generates sparse candidate windows by back-projecting Bag-of-Word image classification scores and assumes objects are more likely to be located by more positive discriminative words. ESS with branch-and-bound search [5] was proposed to reduce the cost in searching subwindow by finding bounds of subwindow scores.

3. Model

The proposed Efficient Maximum Appearance Search (EMAS) model proceeds through four stages to perform large-scale object detection as shown in Fig. 2. During the first stage, we extract multiple complementary features; such as HOG, color moments, etc., for an image, these features are then used to encode the image with a pointwise feature representation during the second stage. In the third stage, we obtain the object confidence maps using a combination of appearance detection models and global context models to look for specific objects within a global context. Finally, the object confidence values are combined to find the highly confident object locations for each object category using maximum subarray search. In the following subsections, we explain in more detail the unique points of our approach, namely, the use of probabilistic prediction over a point ensemble, and the representation, model learning and model inference of the EMAS model. We also extend our model into multi-class categories setting which enables a multi-class object context. Our system can easily adopt multiple feature fusing to boost the performance.

3.1. Probabilistic Prediction over Point Ensemble

Similar to Bag-of-Words like models, where the probabilistic prediction is conducted over the word ensemble contained by the inference body, the EMAS model also estimates the object probabilities using the point ensemble contained within an image area. In particular, let $P(X) = \prod_{i=1}^{n} p(x_i)$ where $P(X)$ is the joint probability over a set of points x_i. The binary discriminative model is used for the figure-ground detection for each object category, which formulates the discriminative probabilities as:

$$\frac{P(X|l=1)}{P(X|l=-1)} = \prod_{i=1}^{n} \frac{p(x_i|l=1)}{p(x_i|l=-1)}, \tag{1}$$

where $l = 1$ denotes the foreground condition and $l = -1$ denotes the background condition respectively. Using the linear discriminative models, e.g. SVM, the logarithm binary discriminative probability can be expressed as:

$$\log(\frac{p(x_i|l=1)}{p(x_i|l=-1)}) = w^T \phi(x_i), \tag{2}$$

where w is the linear weighting vector and $\phi(x)$ denotes the feature expression for a single image point x. Therefore, Eqn. 1 can be formulated into the logarithm form as:

$$\log(\frac{P(X|l=1)}{P(X|l=-1)}) = \sum_{i=1}^{n} w^T \phi(x_i), \tag{3}$$

namely the log-likelihood of an image area to be an object foreground depends on the sum of the pointwise inference in this area.

3.2. Representation: Pointwise Fisher Vector

The performance of the EMAS model relies heavily on the design of pointwise feature representation. In this work,

we choose to extend the Fisher Vector (FV) feature coding method [10] to derive Pointwise Fisher Vector (PFV) coding. Similar to Fisher Vector coding method, the PFV coding uses a Gaussian mixture models (GMMs) $U_\lambda(x) = \sum_{k=1}^{K} \pi_k u_k(x)$ trained on local features of a large image set using Maximum Likelihood (ML) estimation to describe image content. The parameters of the trained GMMs are denoted as $\lambda = \{\pi_k, \mu_k, \Sigma_k, k = 1, \cdots, K\}$, where $\{\pi, \mu, \Sigma\}$ are the prior probability, mean vector and diagonal covariance matrix of Gaussian mixture respectively.

For a local feature x_i extracted from an image, the soft assignments of the descriptor x_i to the kth Gaussian components γ_{ik} is computed by $\gamma_{ik} = \frac{\pi_k u_k(x_i)}{\sum_{k=1}^{K} \pi_k u_k(x_i)}$. The PFV for x_i is denoted as $\phi(x_i) = \{u_{i1}, v_{i1}, \cdots, u_{iK}, v_{iK}\}$ while u_{ik} and v_{ik} is defined as follows:

$$u_{ik} = \frac{1}{\sqrt{\pi_k}} \gamma_{ik} \frac{x_i - \mu_k}{\sigma_k}, v_{ik} = \frac{1}{\sqrt{2\pi_k}} \gamma_{ik} [\frac{(x_i - \mu_k)^2}{\sigma_k^2} - 1]. \quad (4)$$

where σ_k is the square root of the diagonal values of Σ_k. To summarize, we provide a brief analysis of the relationship between FV and PFV coding:

1. PFV extends Fisher Vector Coding [10] to the local feature point level. At each point, the local feature is mapped to GMMs with K Gaussians. The gradient vector with respect to the mean and standard deviation parameters serves as an enriched representation for this local feature. The pointwise representation can also be flexibly merged back to the Fisher Vector global image representation as aforementioned. Compared with VQ, PFV could provide much rich local representation. For VQ, each local feature is mapped to a codebook index while in PFV, x_i is mapped to each GMMs and the gradient vectors enable the local model learning.
2. The pointwise representation $\phi(x_i)$ is sparse since each feature point only has few non-zero GMMs component assignment values γ_{ik}. It means that the model only needs to be applied to these non-zero components in the inference stage thereby making it very efficient. A statistic from SUN09 shows that each local feature is assigned, on average, to 3.5 GMMs components.

3.3. Model Learning

In the training procedure, we assume a series of training samples for one category with bounding boxes window $\{y_1, y_2, ..., y_{n_I}\}$ and corresponding labels $\{l_1, l_2, ..., l_{n_I}\}$. A max-margin formulation is used to learn the linear discriminative model w for each object figure-ground classification. In detail, we formulate the objective function as following:

$$\begin{aligned} w = \quad & \arg\min_{w,\xi} \frac{1}{2}||w||^2 + C \sum_{m=1}^{n_I} \xi_m \quad (5) \\ s.t. \quad & l_m w^T (\frac{1}{Z_m} \sum_{x_i \in y_m} \phi(x_i)) > 1 - \xi_m \\ & \xi_m \geq 0, \forall l_m \in \{1, -1\}, \end{aligned}$$

where $\phi(x_i)$ is the ith pointwise feature in the image area y and we use the ground truth object area as the positive training samples for $l = 1$ and use image areas which have less than 0.4 overlap ratio to the ground truth object areas as the negatve samples for $l = -1$. Normalization factor Z_m is applied to the sum of the pointwise features in order to fit to the SVM optimization. Hard negative mining is done for 3 rounds to enhance the discriminative capability of the model.

3.4. Model Inference

The goal of the EMAS inference step is to find the image area with maximum probability of containing the object,

$$\begin{aligned} \hat{y} &= \arg\max_y \log(\frac{P(X, y|l = 1)}{P(X, y|l = -1)}) \quad (6) \\ &= \arg\max_y \sum_{x_i \in y} w^T \phi(x_i) \\ &= \arg\max_y f(I, y, w) \end{aligned}$$

where $\phi(x_m)$ is the mth pointwise feature in the image area y. We denote an appearance-based detection model as $w = \{w_1^u, w_1^v, \cdots, w_K^u, w_K^v\}$ while w_k^u, w_k^v correspond to the weights for coding vector u_{ik}, v_{ik} respectively. The model scoring function can be generated with the PFV representations $\phi(x_i)$ as follows,

$$f(I, y, w) = \sum_{x_i \in y} \sum_{k=1}^{K} [(w_k^u)^T u_{ik} + (w_k^v)^T v_{ik}], \quad (7)$$

Namely, the scoring of an image area can be substituted by computing score sum of the feature points within the area.

To apply model w on the whole image I and detect high-scored areas, we first extract and encode dense and regularly sampled PFVs– $\phi(x_{ij})$, where $\{i \in [1, N_y], j \in [1, N_x]\}$, N_x and N_y are the sampling point numbers in the width and height direction and $N_x \times N_y = N$ is the total PFV number. Then by computing inner product to all PFVs with the model w, we can produce a rectangle score map M_I, where $M_I(i, j) = w^T \phi(x_{ij})$. In this work, we only consider locating object in rectangle areas $y = [t, b, l, r]$ denoted by the top, bottom, left and right coordinate of the rectangle. Consequently the object detection task is converted to the following optimization problem regarding the scoring function $f(I, y, w)$ in Equation 7. This optimization problem is called 2D maximum subarray sum search:

$$\begin{aligned} \hat{y} &= \arg\max_{y \in \mathcal{Y}} f(I, y, w) \quad (8) \\ f(I, y, w) &= \sum_{i=t}^{b} \sum_{j=l}^{r} M_I(i, j), \end{aligned}$$

where $\mathcal{Y}$ is the rectangle window set within image I. This problem has a number of efficient solutions [19, 9] as compared to simple exhaustive search which has a complexity of $\mathcal{O}(N^2)$. We adopt the method in [19, 21], which

decomposes the search in one dimension to construct efficient dynamic programming problems and has the complexity of $\mathcal{O}(N^{1.5})$. In our experiment, the solution from [19] takes about several milliseconds to search for one confidence map, and the total subarray search for the 107 object categories of SUN09 [20] dataset costs less than one second on one images. Therefore, the computation cost in this subarray search is not a bottleneck of our proposed model.

3.5. Contextual Detection

In this work, we propose a natural way to embed global contextual detection into our detection model. As demonstrated in [2, 22], the object detection performance can be greatly enhanced using the knowledge of global context information in a multi-class setting. The global context is normally the probability values describing how likely the image contains certain object categories, which can provide a reference to the detection results. In our contextual detection, we obtain such probability values from global image classifications. We use the normalized Fisher Vector of the whole image (which can be easily produced from the PFVs) as features. Suppose, there are n_c class in the training dataset, we define the context feature for image I as $\phi_{ctx}(I) = \{c_1, \cdots, c_{n_c}\}$, where c_i are the object existence probability predicted by the ith global classifier. Then, the contextual scoring function is defined as follows,

$$f_{ctx}(I, y, w) \quad = \quad w^T \sum_{x_i \in y} \phi(x_i) + w_{ctx}^T \phi_{ctx}(I), \quad (9)$$

It is worth noting that the contextual detection has several good properties: (1) Stability in the multi-class setting. Normally each context component can depict one attribute of the image, and the weight of the each attribute for detecting certain object can be learned from the training samples. Predictions using additional contextual information is more stable and accurate in problems with large number of object categories and clear object relations. (2) Highly efficient. Defining the global context as the union of classifier outputs is the most efficient way for most recognition models since it requires little additional computation [2, 22]. In our work, the global context can be obtained immediately after running the global image classification.

3.6. Multi-Feature Fusing and Spatial Layout

To effectively model the object appearance, multiple features are often used due to their complementary nature, e.g. HOG or SIFT focus for modeling the local shape, Color Moment for modeling local color statistics, and LBP for modeling the local texture pattern. In the EMAS model, it is easy to fuse multiple features to boost the detection accuracy as well as the effectiveness of the global classification model. We perform independent coding for each kind of local feature. During the training stage, multiple Fisher Vectors are concatenated and fed into the classifier learning.

Table 1: Average running time(s) for 107 classes detection on SUN09.

Total	Fea Extract	PSV Encoding	Model Inference		
			Conf	MaxSearch	Context Det
4.7	0.4	0.7	2.6	0.8	0.2

In the testing stage, multiple features are combined into one confidence map which is then searched efficiently.

We also consider addition of spatial constraints, such as Spatial Pyramid Matching (SPM), into our approach, which will certainly improve the detection accuracy. SPM can be easily added by applying more spatially-structured local models and the maximum subarray search with more complex optimization algorithm. However, at this stage, we concentrate on how to improve the performance with low added-on cost and SPM will bring additional computation cost.

4. Efficiency Analysis

The whole detection process contains three steps, i.e. local feature extraction, PFV encoding, model inference. Here we would like to discuss the detailed efficiency analysis of the last two steps.

PFV encoding includes two parts: soft assignment calculation and the pointwise encoding. The soft assignment has $\mathcal{O}(KND)$ complexity, where N is the number of feature points, K is the number of Gaussians in the GMMs and D is the local feature dimension. The pointwise encoding takes $\mathcal{O}(\mathcal{E}(\gamma_{th})ND)$, where $\mathcal{E}(\gamma_{th})$ represents the average number of GMMs assignments with higher probability than threshold γ_{th} for each feature point. In our experiments, we set $\gamma_{th} = 0.01$ and obtain $\mathcal{E} = 3.5$ on the training image set of SUN09 without losing the performance. Hence the overall computation complexity for PFV coding is near $\mathcal{O}(KND)$ which is equal to the prevalently used Vector Quantization (VQ). For a single computation PFV computes exponential values and products and hence may take more time than square distance of VQ. However, the number of Gaussianse K in PFV is only about hundreds which is much smaller than the codebook size in VQ (from thousands to millions) with similar performance.

The computation in the model inference contains three parts: pointwise confidence mapping, maximum subwindow search and contextual detection. For n_c class, the complexity of pointwise confidence mapping is $\mathcal{O}(n_c\mathcal{E}(\gamma)ND)$. It equals to n_c times inner product of the sparse PSV coding vector. And the maximum subwindow search we adopt has the complexity of $\mathcal{O}(N^{1.5})$ as aforementioned. Finally, compared to the other two parts, the contextual detection cost is trivial since it is only $\mathcal{O}(2n_cKD)$ complexity.

To be more clear, we demonstrate an example computation cost for EMAS in a large scale detection task. The task is performed on SUN09 [20] dataset which includes 107 classes. As shown in Tab. 1, the total cost for 107 classes detection is about 4.7 seconds on a Xeon 2.67GHZ (single

core mode). For one object detector, per category model inference cost is around 0.03 seconds and 3.6 seconds totally for 107 categories. Namely the additional cost for one more detection model is only about 30ms. It proves that the proposed EMAS has high scalability in the number of object categories.

5. Experiments

5.1. Datasets and Metric

We evaluate our proposed EMAS model on two popular datasets, i.e. ImageNet ILSVRC 2012 [28] and SUN09 [20]. ImageNet ILSVRC 2012 is a subset of ImageNet containing 1000 categories and 1.2 million images. In these 1.2 million images, more than 544K images are labeled with object bounding boxes. The validation and test data for this competition consists of 150,000 photographs, collected from flickr and other search engines, hand labeled with the presence or absence of 1000 object categories. A random subset of 50,000 of the images with labels is released as validation data included in the development kit along with a list of the 1000 categories. Our main result is conducted on this validation set since the organizer didn't release the test set annotation after the challenge. The previous experience from the challenge participant shows that the result on validation set is very close to the result on test set [6] since they follows same distribution (validation set is a subset of whole test set.) The evaluation metric is top5 error rate defined by the ILSVRC organizer.

We also use the SUN 09 dataset introduced in [20] for object detection evaluation of 107 object categories, which contains 4,367 training images and 4,317 testing images. SUN 09 [20] has been annotated using LabelMe[27]. The author also annotated an additional set of 26,000 objects using Amazon Mechanical Turk to have enough training samples for the baseline detectors [2]. These detectors span from regions (e.g., road, sky, buildings) to well defined objects (e.g., car, sofa, refrigerator, sink, bowl, bed) and highly deformable objects (e.g., river, towel, curtain). The employed evaluation metric is Average Precision (AP) and mean of AP (mAP).

5.2. Implementation Details

We first normalize the image by setting the longest edge of the image to 500 pixels. Afterwards, we extract two kinds of low-level features for all the experiments. The first one is dense SIFT feature from VL-Feat [32] using multiple scales setting (spatial bins are set as 4, 6, 8, 10) with 6 pixel step. The second one is the local color comment (CM) proposed in [10]. These two features have been shown to be complementary to each other for the task of object classification [10]. Each SIFT and CM feature is reduced to 60 dimensions for noise removal. The number of mixtures in the GMMs model in PSV coding is set to 128 for SUN dataset and 256 for ILSVRC dataset. We sample 500,000 descriptors from the training images of ILSVRC and perform EM to obtain the GMMs. For all experiments, we only output the maximum subwindow for one image per class at testing stage, namely we use a precision-preferred detector. Multiple detections can be obtained by iteratively performing the EMAS on one image. All the experiments are conducted on a Xeon Server with 32GB memory using single core mode.

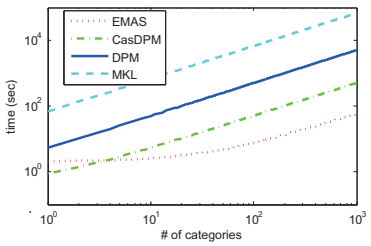


Figure 3: Inference cost comparison in a multi-class setting.

For model learning, we fix the parameter C of SVM as 1 for all experiments. The hard training constraint is mined in a manner similar to the model inference step described earlier except that we restrict the number of output windows to 30 for one image followed by a Non-Maximum Suppression step. The total training process usually takes about half an hour for one class.

5.3. Efficiency Comparison

We compare the computational cost of EMAS with three other object detection models in a multi-class setting: 1) Multiple kernel learning for object detection (MKL) [16] using three-stage linear and non-linear detection, 2) Deformable Part Model [2], and 3) Cascade DPM [4].

Figure 3 shows the computation cost of the various approaches on the ILSVRC 2012 dataset with a varying number of object categories. EMAS, on an average, takes about 58.4 seconds to process an image, which consists of about 1.9 seconds for feature extraction and feature encoding, and about 56.5 seconds for model inference for 1000 categories. So, per object category cost, on an average, is 56ms. In contrast, for both CasDPM and DPM, the feature pyramid takes about 375ms. And it takes 500ms and 5s respectively for model inference per category per image(may change for different setting). Additionally, the cost for MKL reported in [16] is 67 seconds per category for one image. Therefore, we can infer that EMAS is at least one order of magnitude faster as compared to other approaches for large-scale object detection. When number of categories is small, however, it can be observed that EMAS is not the fastest due to the cost of feature encoding.

5.4. Performance Evaluation

5.4.1 Large Scale Object Detection on ILSVRC2012:

ILSVRC2012 is a large challenging dataset including 1000 object categories. We first perform the classification task to obtain the object context. For each category, we train a one-vs-all classifier using an implementation of stochastic dual-form SVM solver [30]. The top 5 error ratio ($error_{cls}$) using two features is 0.326 which is very close to the pub-

Table 2: Object classification and detection results on ILSVRC 2012.

	XRCE/INRIA	Oxford_DPM	Oxford_Mix	ISI_CasDPM	EMAS
GMMs size	256	1024	1024	256	**256**
Multi-Fea+SPM	2 fea	2 fea	2 fea	4 fea+SPM	**2 fea**
$error_{cls}$	0.334	0.269	0.269	**0.261**	0.326
$error_{det}$	n.a.	0.529	**0.500**	0.536	0.554
acc_{det}	n.a.	0.644	**0.684**	0.628	**0.662**

lic result 0.334 from **XRCE/INRIA** in the challenge with similar setting. The result using single dense SIFT feature is 0.380. The complementary effect from CM improves the overall performance. It is worth noting that our performance can be further boosted with large GMM for FV. e.g. **Oxford** gets 0.269 when sets the size as 1024 which is 4 times larger than our implementation. We train our detection using the same SVM solver. The initialization of the detection model is trained using the object feature and a large amount of negative images. 3 round of hard sample mining is utilized.

For detection, we compare our results with the challenging entries [1]: (1) **Oxford_DPM** is the result from DPM detection over baseline classification scores. (2) **Oxford_Mix** used the detection result from DPM and retrain the foreground model with complicated classification model which also is the best result from **Oxford**. (3) **ISI_CasDPM** is the result using cascade object detection with deformable part models, restricting the sizes of bounding boxes. We show the comparison results on ILSVRC2012 dataset in Tab. 2. Our detection result $error_{det}$ reaches 0.554 top 5 error rate which is comparable to the DPM and CasDPM while the single feature result using SIFT only is 0.582. Moreover, it is worth noting that the detection result of ILSVRC2012 heavily relies on the performance of classification. Usually, detection will be performed to the top ranked image with high classification confidence, i.e. a combination of two steps: first classifier the right categories and then perform the localization. Thus the error rate can be approximately interpreted as $error_{det} = 1 - (1 - error_{cls}) * acc_{det}$ where the acc_{det} shows the real detection accuracy for each detection model. We show the acc_{det} in Tab. 2. It can be seen that our localization ability of our detection model is also comparable to the state-of-the-art model.

5.4.2 Multi-Label Object Detection on SUN09:

SUN09 is a very challenging datasets with rich contextual information. The concerned object categories span from regions (e.g., road, sky, buildings) to well defined objects (e.g., car, sofa, sink, bowl, bed) and highly deformable objects (e.g., river, towel, curtain). We first trained the global object classification model. Each class is trained independently using linear SVM. The mAP of the classifiers is about 29.6% for 107 classes on SUN09 dataset. The classification scores on the training set is obtained by 10-fold cross validation. We perform the proposed EMAS detection model on the 107 classes and compare with the DPM. We use the results of DPM on SUN09 released by the author of [20] which is 7.06% mAP for 107 objects. Further [20] refines this baseline result by modeling the co-occurrence and relative spatial relation of objects with a tree graphical model and obtain the improvement to 8.37% mAP. Our base detector without contextual training obtains 7.26% mAP which is slightly better than the result of DPM and we obtain 8.44% mAP with our contextual detection. Our outperformed categories are also on the highly deformable objects. In Section. 5.4.3, we will provide a more comprehensive analysis on this feature of the EMAS model.

[1] www.image-net.org/challenges/LSVRC/2012/results.html

Table 3: Object detection result on Sun09(AP %).

	plane	bed	bkcase	building	closet	field	floor	grass	mountain	river	
DPM[2]	**35.1**	26.3	2.3	**14.4**	1.1	**19.8**	31.3	11.0	17.2	2.9	
EMAS	12.7	**34.1**	**14.8**	14.3	**12.8**	18.9	**38.1**	**12.3**	**25.6**	**12.4**	
	road	sea	shelves	showcase	sky	sofa	toilet	tree	wall	water	mAP
DPM[2]	33.2	28.7	2.6	0.0	55.3	11.5	**22.0**	10.9	14.7	1.5	17.1
EMAS	**34.9**	**35.0**	**13.6**	**11.9**	**61.9**	**12.7**	11.7	**12.4**	**21.9**	**15.1**	**21.4**

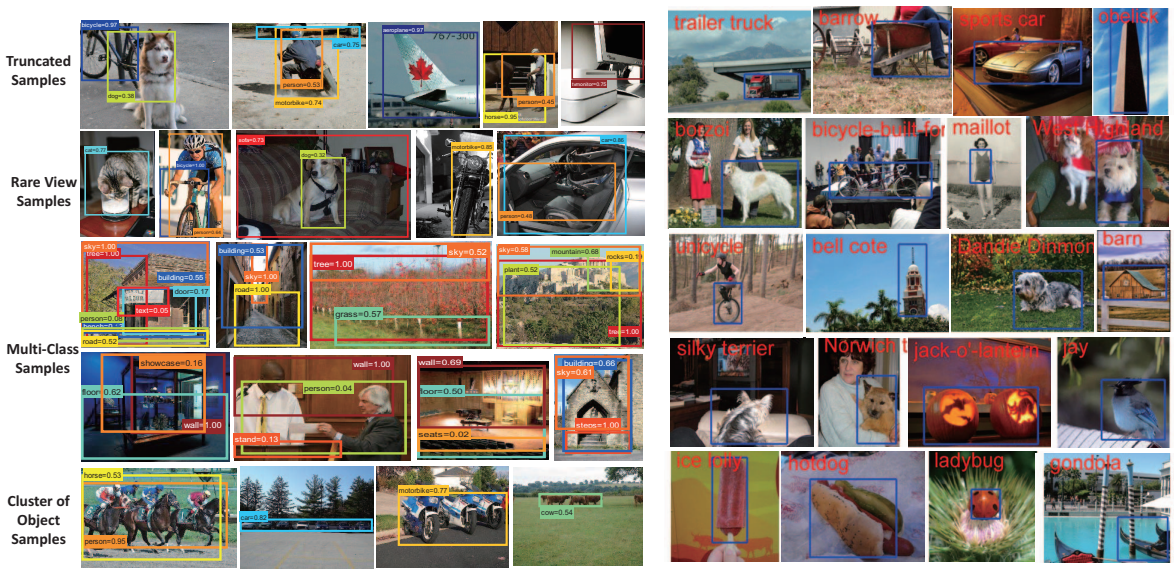
5.4.3 Object Detection with Large Appearance Variance

Our appearance-based model is appealing for object detection with large variation of appearance. Here, we show 20 classes amorphous object detection result from SUN09 and compare with the DPM [2] in Tab. 3. These classes range from 1) regions (e.g. sky, building, road, river) and 2) objects with large shape variation (e.g. bed, sofa, shelves, aeroplane). The EMAS achieves better results. There are some interesting features of EMAS revealed by some example detection shown in Fig. 4. The model is purely appearance-based, i.e with no shape constraint, thus the algorithm is good at handling truncated/occluded objects (Fig. 4a, 1st row, such as part of cars and bicycles), rare view objects (Fig. 4, 2nd row, such strange view of cats, sofa, motorbikes) and detecting region objects (Fig. 4, 3rd and 4th row, such as sky, buildings, trees, floor). But it also causes the problem that it can not distinguish one object from a cluster of objects (e.g. a cluster of horses, cars, cows, shown in Fig. 4a, 5th row).

We show some sample detection results from ILSVRC2012 in Fig. 4b, the large number of categories creates large diversity in the object categories. It is interesting to see that the proposed detector can detect the object in the 1000 categories pool. We plot more results in the supplementary files.

6. Conclusion

In this paper, we designed an efficient large-scale object detection approach by extending Fischer Vector encoding to the point-level. This enabled us to transform the object detection problem into a problem of searching for a sub-window with the maximum sum leading to an order of magnitude of speed-up over the state-of-the-art approaches while maintaining comparable accuracy on the major large-scale object detection benchmarks. It is our belief that this significant speed-up makes large-scale object detection



(a) Sample results from SUN09

(b) Sample results from ILSVRC2012

Figure 4: Sample results. Best viewed in the enlarged color pdf file.

practical. Moreover, the proposed approach could further integrate global object contextual information into the detection model with little extra computational cost, which may make it very effective for object detection under difficult conditions, such as occluded objects. In future work, we plan to explore the possibility of incorporating the spatial layout information, such as SPM, in an efficient manner.

Acknowledgment

This research is supported by the Singapore National Research Foundation under its International Research Centre @Singapore Funding Initiative and administered by the IDM Programme Office.

References

[1] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)

[2] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object Detection with Discriminatively Trained Part-Based Models. IEEE Trans. Pattern Analysis and Machine Intelligence (2010)

[3] Chum, O., Zisserman, A.: An Exemplar Model for Learning Object Classes. In: CVPR. (2007)

[4] Felzenszwalb, P., Girshick, R.: Cascade object detection with deformable part models. In: CVPR. (2010)

[5] Lampert, C., Blaschko, M., Hofmann, T.: Beyond sliding windows: Object localization by efficient subwindow search. In: CVPR (2008)

[6] N. Gunji, T. Higuchi, K. Yasumoto, H. Muraoka, Y. Ushiku, T. Harada, and Y. Kuniyoshi.: Scalable Multiclass Object Categorization with Fisher Based Features http://www.image-net.org/challenges/LSVRC/2012/isi.pdf

[7] Lampert, C., Blaschko, M.: Learning to Localize Objects with Structured Output Regression. In: ECCV. (2008)

[8] An, S., Peursum, P., Liu, W., Venkatesh, S.: Efficient subwindow search with submodular score functions. In: CVPR. (2011)

[9] An, S., Peursum, P., Liu, W., Venkatesh, S.: Efficient algorithms for subwindow search in object detection and localization. In: CVPR. (2009)

[10] Florent Perronnin, J.S., Mensink, T.: Improving the Fisher Kernel for Large-Scale Image Classification. In: ECCV. (2010)

[11] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision (2010)

[12] Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: CVPR. (2009)

[13] Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: CVPR. (2011)

[14] Yang, J., Yu, K., Gong, Y.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. (2009)

[15] Wang, J., Yang, J., Yu, K., Lv, F., Huang, T.: Locality-constrained linear coding for image classification. In: CVPR. (2010)

[16] Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple Kernels for Object Detection. In: ICCV. (2009)

[17] Chatfield, K., Lempitsky, V., Vedaldi, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: BMVC. (2011)

[18] Zhou, X., Yu, K., Zhang, T.: Image classification using super-vector coding of local image descriptors. In: ECCV. (2010)

[19] Bentley, J.: Programming Pearls (2nd Edition). Addison-Wesley Professional (1999)

[20] Choi, M.J., Lim, J., Torralba: Exploiting hierarchical context on a large database of object categories. In: CVPR. (2010)

[21] Lempitsky,V., Zisserman, A., Learning to Count Objects in Images In: NIPS. (2010)

[22] Song, Z., Chen, Q., Huang, Z., Hua, Y., Yan, S.: Contextualizing object detection and classification. In: CVPR. (2011)

[23] Zhu, L., Chen, Y., Yuille, A.: Learning a Hierarchical Deformable Template for Rapid Deformable Object Parsing. IEEE Trans. Pattern Analysis and Machine Intelligence (2010)

[24] Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet Classification with Deep Convolutional Neural Networks. In: NIPS. (2012)

[25] Csurka G., Perronnin F.: A Simple High Performance Approach to Semantic Segmentation. In: BMVC. (2008)

[26] Fischler, M.A., Elschlager, R.A.: The Representation and Matching of Pictorial Structures IEEE Trans. Computers. (1973)

[27] Russell, B., Torralba, A., Murphy, K.: LabelMe: a database and web-based tool for image annotation. International Journal of Computer Vision (2008)

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei,: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR. (2009).

[29] Dubout, C., Fleuret, F.: Exact Acceleration of Linear Object Detectors. In: ECCV. (2012)

[30] Hsieh, C., Chang, K., Lin, C., Keerthi, S., Sundararajan, S.: A dual coordinate descent method for large-scale linear SVM. In: ICML. (2008)

[31] Song, H., Zickler S.,Althoff T., Girshick R., Fritz M., Geyer C., Felzenszwalb P., Darrell T.: Sparselet Models for Efficient Multiclass Object Detection In: ECCV. (2012)

[32] Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008.