ConceptMap: Mining Noisy Web Data for Concept Learning

Eren Golge and Pinar Duygulu

Bilkent University, 06800 Cankaya, Turkey

Abstract. We attack the problem of learning concepts automatically from noisy Web image search results. The idea is based on discovering common characteristics shared among subsets of images by posing a method that is able to organise the data while eliminating irrelevant instances. We propose a novel clustering and outlier detection method, namely Concept Map (CMAP). Given an image collection returned for a concept query, CMAP provides clusters pruned from outliers. Each cluster is used to train a model representing a different characteristics of the concept. The proposed method outperforms the state-of-the-art studies on the task of learning from noisy web data for low-level attributes, as well as high level object categories. It is also competitive with the supervised methods in learning scene concepts. Moreover, results on naming faces support the generalisation capability of the CMAP framework to different domains. CMAP is capable to work at large scale with no supervision through exploiting the available sources.

Keywords: Weakly-labelled data, Clustering and outlier detection, Semi-supervised model learning, ConceptMap, Attributes, Object detection, Scene classification.

1 Introduction

The need for manually labelled data continues to be one of the most important limitations in large scale recognition. Alternatively, images are available on the Web in huge amounts. This fact recently attracted many researchers to build (semi-)automatic methods to learn from web data collected for a given concept. However, there are several challenges that makes the data collections gathered from web different from the hand crafted datasets. Images on the web are "in the wild" inheriting all types of challenges due to variations and effects. Since usually images are gathered based on the surrounding text, the collection is very noisy with several visually irrelevant images as well as variety of images corresponding to different characteristic properties of the concept (Figure1).

For the queried data for automatic learning of concepts, we propose a novel method to obtain a representative groups with irrelevant images removed. Our intuition is that, given a concept category by a query, although the list of images returned include irrelevant ones, there will be common characteristics shared among subset of images. Our main idea is to obtain visually coherent subsets, that are possibly corresponding to semantic sub-categories, through clustering and to build models for each sub-category (see Figure2). The model for each concept category is then a collection of multiple models, each representing a different aspect.

D. Fleet et al. (Eds.): ECCV 2014, Part VII, LNCS 8695, pp. 439–455, 2014.
© Springer International Publishing Switzerland 2014

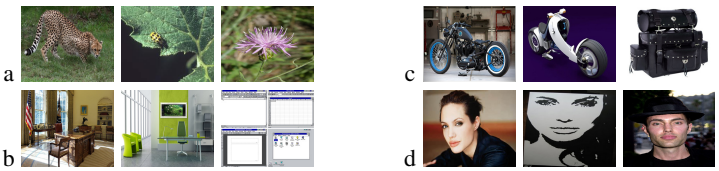


Fig. 1. Example Web images collected for query keywords (a) spotted, (b) office, (c) motorbikes, (d) Angelina Jolie. Even in the relevant images, the concepts are observed in different forms requiring grouping and irrelevant ones to be eliminated.

To retain only the relevant images that describe the concept category correctly, during clustering we need to remove outliers, i.e. irrelevant ones. The outliers may resemble to each other while not being similar to the correct category resulting in a **outlier cluster**. Alternatively, outlier images could be mixed with correct category images inside **salient clusters** corresponding to relevant ones. These images, that we refer to as **outlier elements**, should also be removed for the quality data for learning.

We propose a novel method **Concept Maps (CMAP)** for which organises the data by purifying it not only from outlier clusters but also from outlier elements in salient clusters. CMAP captures category characteristics through organising the set of given instances into sub-categories pruned from irrelevant instances. It is a generic method that could be applied on any type of concept from low-level attributes to high level object and scene categories as well as faces.

Contributions:

- We attack the problem of building a general framework to learn visual concepts by only query concept, through exploiting large volumes of weakly labelled data on the web.
- Unlike most of the recent studies that focus on learning specific types of categories from noisy images downloaded from web (such as objects [17,33], scenes[55], attributes[53,18], and faces [2,43,20]) we propose a general framework which is applicable to many domains from low level attributes to high level concepts.
- We aim to learn models that have the ability to categorise images and regions across datasets without being limited to a single source of data.
- As in [33,5] we address three main challenges in learning visual concepts from noisy web results: (i) **Irrelevant images** returned by the search engines due to keyword based queries on the noisy textual content. (ii) **Intra-class variations** within a category resulting in multiple groups of relevant images. (iii) **Multiple senses** of the concept. (5) We aim to answer not only ”which concept is in the image?”, but also ”where the concept is?” as in [5] . Local patches are considered as basic units to solve the localisation as well as to eliminate background regions.
- We use only visual informations extracted from the images gathered for a given query word, and do not require any other additional knowledge such as surrounding text, metadata or GPS-tags [48,3,23].
- The collection returned from web is used in its **pure** form without requiring any prior supervision (manual or automatic) for organisation of the data [3,48,33].

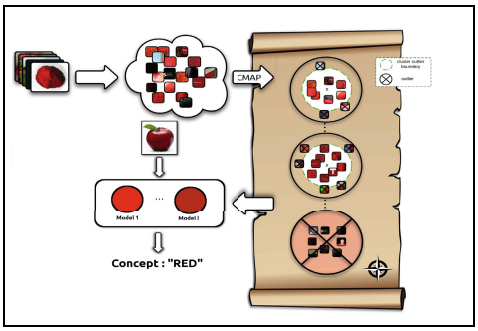


Fig. 2. Overview of our framework for concept learning shown on example concept "Red". Images are collected from web for a given keyword. Concept Map (CMAP) organises the data into clusters which are pruned from outlier elements inside salient clusters and outlier clusters. each cluster is then used as a sub-model for learning and localising the concept in a given image possibly from a different collection.

2 Related Work

Our work is related to several studies in the literature from different perspectives. We try to discuss the most relevant ones by grouping them into three categories. Reviewing the huge literature on object and scene recognition is far from the scope of this study.

Learning Attributes: The use of attributes has been the focus of many recent studies [15,29,6]. Most of the methods learn attributes in a supervised way [16,31] with the goal of describing object categories. Not only semantic attributes, but classemes [52] and implicit attributes [46] have also been studied. We focus on attribute learning independent of object categories and learn different intrinsic properties of semantic attributes through models obtained from separate clusters that are ultimately combined in a single semantics. Learning semantic appearance attributes, such as colour, texture and shape, on ImageNet dataset is attacked in [47] relying on image level human labels using AMT for supervised learning. We learn attributes from real world images collected from web with no additional human effort for labelling. Another study on learning colour names from web images is proposed in [53] where a pLSA based model is used for representing the colour names of pixels. Similar to ours, the approach of Ferrari and Zisserman [18] considers attributes as patterns sharing some characteristic properties where basic units are the image segments with uniform appearance. We prefer to work on patch level alternative to pixel level which is not suitable for region level attributes such as texture; image level which is very noisy; or segment level which is difficult to obtain clearly.

Learning Object Categories from Noisy Web Data: Several recent studies tackle the problem of building qualified training sets by using images returned from image search engines[17,3,1,13,33,48]. Fergus et al. [17] propose a pLSA based method in which the spatial information is also incorporated in the model. They collected noisy images from

Google as well as a validation set which consists of top five images collected in different languages which was used to pick the best topics. They experimented classification on subsets of Caltech and Pascal datasets, and re-ranking of Google results. The main drawback of the method is the dependency to the validation set. Moreover, the results indicate that the variations in the categories are not handled well. Berg and Forsyth [3] use visual features and surrounding the text for collecting animal images from web. Visual exemplars are obtained through clustering text. They require the relevant clusters to be identified manually, as well as an optional step of eliminating irrelevant images in clusters. Note that these two steps are automatically performed in our framework. Li and Fei-Fei [33] presents the OPTIMOL framework for incrementally learning object categories from web search results. Given a set of seed images a non parametric latent topic model is applied to categorise collected web images. The model is iteratively updated with the newly categorised images. To prevent over specialised results, a set of cache images with high diversity are retained at each iteration. While the main focus is on the analysis of the generated collection, they also compared the learned models on the classification task on the dataset provided in [17]. The validation set is used to gather the seed images. The major drawback of the method is the high dependency to the quality of the seed images and the risk for concept drift during iterations. Schroff et al. [48] first filters out the abstract images (drawings, cartoons, etc.) from the resulting set of images collected through text and image search in Google for a given category. Then, they use text and metadata surrounding the images to re-rank the images. Finally they train a visual classifier by sampling from the top ranked images as positives and random images from other categories as negatives. Their method highly depends on the filtering and text-based re-ranking as shown with the lower performances obtained by visual only based classifier. Berg and Berg [1] find iconic images that are the representatives of the collection given a query concept. First they select the images with objects are distinct from background. Then, the high ranked images are clustered using k-medoids to consider centroid images as iconic. Due to the elimination of several images in the first step it is likely that helpful variations in the dataset are removed. Moreover, clustering does not handle the images in outlier clusters to be chosen as iconic. NEIL is the most similar study to ours [5]. Similar to CMAP in NEIL multiple sub-models are learned automatically for each concept. It works on attributes, objects and scenes as well and localises objects in the images. CMAP differentiates from NEIL in some aspects. We also perform experiments on recognition of faces which was not handled in NEIL. Unlike NEIL where a single type of representation is used to describe attributes, objects and scenes, we use different descriptors for each although not specialised and can be replaced with others. However, this distinction also allows us to consider faces and possibly the videos in the future. The second difference lies in the organisation of the data for learning sub-models. High computational power required for NEIL, as well as the additional knowledge discovered and the iterative process required for learning makes it difficult to compare.

Learning Face Name Associations: Learning the faces associated with the name has been studied recently[2,43,19,21,20,42,49]. We focus on the task of learning faces given a single query name. Unlike [43,19] where a single densest component is sought in the

similarity graph - which corresponds to the most similar subset of faces-, we seek for multiple subgroups that represents different characteristics of the people.

Learning Discriminative Patches: Our method is also related to the recently emerged studies in discovering discriminative patches. [34] [50] [27] [50,9,8,25,10,26,35,7]. In these studies weakly labeled datasets are leveraged for learning visual patches that are representative and discriminative. We aim to discover the patches or the entire images representing the collected data in the best way. However, we also want to keep the variations in the concept for allowing intra-class variations and multiple senses to be modelled through different sub-groups. We want to learn the characteristics of the concepts independent of other concepts, and don't consider discriminative characteristics.

3 Concept Maps

We propose CMAP which is inspired from the well-known Self Organizing Maps (SOM) [28]. In the following, SOM will be revisited briefly, and then CMAP will be described.

Revisiting Self Organizing Maps (SOM): Intrinsic dynamics of SOM are inspired from developed animal brain where each part is known to be receptive to different sensory inputs and which has a topographically organized structure[28]. This phenomena, i.e. "receptive field" in visual neural systems [24], is simulated with SOM, where neurons are represented by weights calibrated to make neurons sensitive to different type of inputs. Elicitation of this structure is furnished by competitive learning approach.

Consider input $X = \{x_1, .., x_M\}$ with M instances. Let $N = \{n_1, ..., n_K\}$ be the locations of neuron units on the SOM map and $W = \{w_1, ..., w_K\}$ be the associated weights. The neuron whose weight vector is most similar to the input instance x_i is called as the winner and denoted by $\hat{v}$. Weights of the winner and units in the neighbourhood are adjusted towards the input at each iteration t with delta learning rule.

$$w_j^t = w_j^{t-1} + h(n_i, n_{\hat{v}} : \epsilon^t, \sigma^t)[x_i - w_j^{t-1}] \tag{1}$$

Update step is scaled by the window function $h(n_i, n_{\hat{v}} : \epsilon^t, \sigma^t)$ for each SOM unit, inversely proportional to the distance to the winner (Eq.2). Learning rate ϵ is a gradually decreasing value, resulting in larger updates at the beginning and finer updates as the algorithm evolves. σ^t defines the neighbouring effect so with the decreasing σ, neighbour update steps are getting smaller in each epoch. Note that, there are different alternatives for update and windows functions in SOM literature.

$$h(n_i, n_{\hat{v}} : \epsilon^t, \sigma^t) = \epsilon^t \exp \frac{-||n_j - n_{\hat{v}}||^2}{2\sigma^{t^2}} \tag{2}$$

Clustering and Outlier Detection with CMAP: We introduce excitation scores $E = \{e_1, e_2, \ldots, e_K\}$ where e_j, the score for neuron unit j, is updated as in Eq.3.

$$e_j^t = e_j^{t-1} + \rho^t(\beta_j + z_j) \tag{3}$$

As in SOM, window function is getting smaller with each iteration. z_j is the activation or win count for the unit j, for one epoch. ρ is learning solidity scalar that represents the decisiveness of learning with dynamically increasing value, assuming that later stages of the algorithm has more impact on the definition of salient SOM units. ρ is equal to the inverse of the learning rate ϵ. β_j is the total measure of the activation of jth unit in an epoch, caused by all the winners of the epoch but the neuron itself (Eq.4).

$$\beta_j = \sum_{\hat{v}=1}^{u} h(n_j, n_{\hat{v}}) z_{\hat{v}} \tag{4}$$

At the end of the iterations, normalized e_j is a quality value of a unit j. Higher value of e_j indicates that total amount of excitation of the unit j in whole learning period is high thus it is responsive to the given class of instances and it captures notable amount of data. Low excitation values indicate the contrary. CMAP is capable of detecting outlier units via a threshold θ in the range $[0, 1]$.

Let $C = \{c_1, c_2, \ldots, c_K\}$ be the cluster centres corresponding to each unit. c_j is considered to be a **salient cluster** if $e_j \geq \theta$, and an **outlier cluster** otherwise.

The excitation scores E are the measure for saliency of neuron units in CMAP. Given the data belonging to a category, we expect that data is composed of sub-categories that share common properties. For instance `red` images might include tones to be captured by clusters but they are supposed to share a common characteristics of being red. For the calculation of the excitation scores we use individual activations of the units as well as the neighbouring activations. Individual activations measure being a salient cluster corresponding to a particular sub-category, such as `lighter red`. Neighbourhood activations count the saliency in terms of the shared regularity between sub-categories. If we don't count the neighbourhood effect, some unrelated clusters would be called salient, e.g. noisy white background patches in `red` images.

Outlier instances in salient clusters (**outlier elements**) should also be detected. After the detection of outlier neurons, statistics of the distances between neuron weight w_i and its corresponding instance vectors is used as a measure of instance divergence. If the distance between the instance vector x_j and its winner's weight $\hat{w}_i$ is more than the distances of other instances having the same winner, x_j is raised as an outlier element. We exploit box plot statistics, similar to [39]. If the distance of the instance to its cluster's weight is more than the upper-quartile value, then it is an outlier. The portion of the data, covered by the upper whisker is decided by τ.

CMAP provides good basis of cleansing of poor instances whereas computing cost is relatively smaller since an additional iteration after clustering phase is not required. All the necessary information (excitation scores, box plot statistics) for outliers is calculated at runtime of learning. Hence, CMAP is suitable for large scale problems.

CMAP is also able to estimate number of intrinsic clusters of the data. We use PCA as a simple heuristic for that purpose, with defined variance ν to be retained by the selected first principle components. Given data, principle components describing the data with variance ν is used as the number of clusters for the further processing of CMAP. If we increase ν, CMAP latches more clusters.

$$Num.Clusters = \max_{q} \left(\frac{\sum_{i=1}^{q} \lambda_i}{\sum_{j=1}^{p} \lambda_j} \leq \nu \right) \tag{5}$$

q is the number of top principle components selected after PCA and p is the dimension of instance vectors. λ is the eigenvalue of corresponding component.

Discussion of Other Methods on Outlier Detection with SOM: [37,38] utilise the habitation of the instances. Frequently observed similar instances excites the network to learn some regularities and divergent instances are observed as outliers. [22] benefits from weights prototyping the instances in a cluster. Thresholded distance of instances to the weight vectors are considered as indicator of being outlier. In [56], aim is to have different mapping of activated neuron for the outlier instances. The algorithm learns the formation of activated neurons on the network for outlier and inlier items with no threshold. It suffers from the generality, with its basic assumption of learning from network mapping. LTD-KN [51] performs Kohonen learning rule inversely. An instance activates only the winning neuron as in the usual SOM, but LTD-KN updates winning neuron and its learning windows decreasingly.

These algorithms only eliminate outlier instances ignoring outlier clusters unlike CMAP. Another difference of CMAP is the computation cost. Most of outlier detection algorithms model the data and iterate over the data again to label outliers. CMAP has the ability to detect outlier clusters and the items in the learning phase. Thus, there is no need for a further iteration, it is all done in a single pass in our method.

Algorithm 1. CMAP

In the real code we use vectorized implementation whereas we write down iterative pseudo-code for the favour of simplicity.
Input: $X, \theta, \tau, K, T, \nu, \sigma^{init}, \epsilon^{init}$
Output: $OutlierUnits, Mapping, W$
set each item z_i in Z to 0
$u \leftarrow estimateUnitNumber(X, variation)$
$W \leftarrow randomInit(u)$
while $t \leq T$ **do**
  $\epsilon^t \leftarrow computeLearningRate(t, \epsilon^{init})$
  $\rho^t \leftarrow 1/\epsilon^t$
  set each item β_i in B to 0
  select a batch set $X^t \subset X$ with K instances
  for *each* $x_i \in X$ **do**
    $\hat{w}_i^t \leftarrow findWinner(x_i, W)$
    $\hat{v} \leftarrow min_j(||x_i - w_j||)$
    increase win count $z_{\hat{w}^t} \leftarrow z_{\hat{w}_i^t} + 1$
    increase win count $z_{\hat{v}} \leftarrow z_{\hat{v}} + 1$
    for *each* $w_k \in W$ **do**
      $\beta_k^t = \beta_k^t + h(n_k, n_{\hat{v}})$
      $w_k = w_k + h(n_k, n_{\hat{v}})||x_i - w_{\hat{v}}||$
    end
  end
  for *each* $w_j \in W$ **do**
    $e_j^t = e_j^{t-1} + \rho^t(\beta_j^t + z_j)$
  end
  $t \leftarrow t + 1$
end
$W_{out} \leftarrow thresholding(E, \theta)$
$W_{in} \leftarrow W \setminus W_{out}$
$Mapping \leftarrow findMapping(W_{in}, X)$
$Whiskers \leftarrow findUpperWhiskers(W_{in}, X)$
$X_{out} \leftarrow findOutlierIns(X, W_{in}, Whiskers, \tau)$
return $W_{out}, X_{out}, Mapping, W$

4 Concept Learning with CMAP

We utilise the clusters, that are obtained through CMAP as presented above, for learning sub-models in concepts. We exploit the proposed framework for learning of attributes, scenes, objects and faces. Each task requires the collection of data, clustering and outlier

detection with CMAP, and training of sub-models from the resulting clusters. In the following, first we will describe the attribute learning, and then describe the differences in learning other concepts. Implementation details are presented in Section5.

Learning Low-Level Attributes: Most of the methods require learning of visual attributes from labelled data, and cannot eliminate human effort. Here, we describe our method in learning attributes from web data without any supervision.

We collect web images through querying colour and texture names. The data is weakly labelled, with the labels given by queries. Hence, there are irrelevant images in the collection, as well as images with a tiny portion corresponding to the query keyword.

Each image is densely divided into non-overlapping fixed-size patches to sufficiently capture the required information. We assume that the large volume of the data itself is sufficient to provide instances at various scales and illuminations, and therefore we did not perform any scaling or normalisation. The collection of all patches extracted from all images for a single attribute is then given to CMAP to obtain clusters which are likely to capture different characteristics of the attribute as removing the irrelevant image patches.

Each cluster obtained through CMAP is used to train a separate classifier. Positive examples are selected as the members of the cluster and negative instances are selected among the outliers removed by CMAP and also elements from other categories.

Learning Scene Categories: To show CMAP capability on higher level concepts, we target scene categories. In this case, we use the entire images as instances, and aim to discover groups of images each representing a different property of the scene, at the same time by eliminating the images that are either spurious. These clusters are then used as models similar to the attribute learning.

Learning Object Categories: In the case of objects, we detect salient regions on each image via [11], to eliminate background noise. Then these salient regions are fed into CMAP framework for clustering.

Learning Faces: We address the problem of learning faces associated with a name -which is generally referred to face naming in the literature-, through finding salient clusters in the set of images collected from web through querying the name. Here, the clusters are likely to correspond to different poses and possibly different hair and make-up style differences as well as ageing effects. Note that this task is not the detection of faces, but recognition of faces for a given name. We detect the faces in the images, and only use a single face with the highest confidence for each image.

5 Experiments

5.1 Qualitative Evaluation of Clusters

As Figure3 depicts, CMAP captures different characteristics of concepts in separate salient clusters, while eliminating outlier clusters that group irrelevant images coherent among themselves, as well as outlier elements wrongly mixed with the elements of salient clusters . On more difficult tasks of grouping objects and faces, CMAP is again successful in eliminating outlier elements and outlier clusters as shown in Figure4.

Fig. 3. For colour and texture attributes *brown* and *vegetation* and scene concept *bedroom*, randomly sampled images detected as (i) elements of **salient clusters**, (ii) elements of **outlier clusters**, and (iii) **outlier elements** in salient clusters. CMAP detects different shades of "Brown" and eliminates some superiors elements belonging the different colors. For the "Vegetation" and "Bedroom", CMAP again divides the visuals elements with respect to structural and angular properties. Especially for "bedroom", each cluster is able to capture different view-angle of the images as it successfully removes outlier instances with some of little mistakes that are belonging to the label but not representative (circular bed in very shiny room) for the concept part.

5.2 Implementation Details

Parameters of CMAP are tuned on a small held-out set gathered for each concept class for color, texture, and scene. Best ν is selected by the optimal Mean Squared Error and threshold parameters are tuned by cross-validation accuracies. Figure5 depicts the effect of parameters θ, τ and ν. For each parameter the other two are fixed at the optimum value.

We use LINLINEAR library [14] for L1 norm SVM classifiers. SVM parameters are selected with 10-fold cross validation.

CMAP implementation is powered by CUDA environment. Matrix operations observed for each iteration is kernelized by CUDA codes. It provides good reduction in time, especially if the instance vectors are long and the data is able to fit into GPU memory. Hence, we are able to execute all the optimization in GPU memory. Otherwise some dispatching overhead is observed between GPU and global memory that sometimes hinge the efficiency.

5.3 Attribute Learning

Datasets and Representation: We collected images from Google for 11 distinct colours as in [53] and 13 textures. We included the terms "colour" and "texture" in the queries, such as "red colour", or "wooden texture". For each attribute, 500 images are collected. In total we have 12000 web images. Each image is divided into 100x100 non-overlapping patches. Unlike [53], we didn't apply gamma correction. For colour concepts we use 10x20x20 bins Lab colour histograms and for texture concepts we use BoW representation for densely sampled SIFT [36] features with 4000 words. We keep

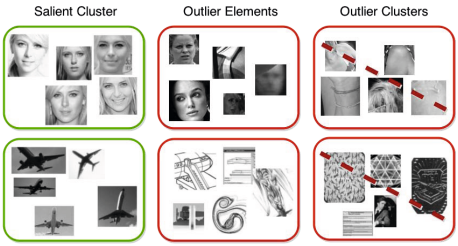


Fig. 4. CMAP results for object and face examples. Left columns shows one example of salient cluster. Middle column shows outlier instances captured from salient clusters. Right column is the detected outlier clusters.

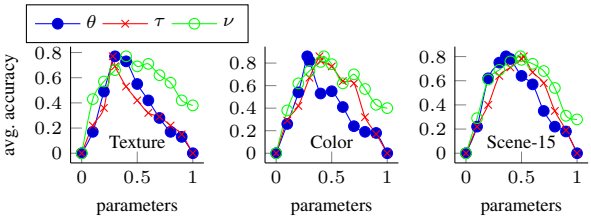


Fig. 5. Effect of parameters on average accuracy. For each parameter, the other two are fixed at their optimal values. θ is outlier cluster threshold, ν is PCA variation used for the estimation of number of clusters,τ is the upper whisker threshold for the outliers in salient clusters.

the feature dimensions high to utilise from the over-complete representations of the instances with L1 norm linear SVM classifier.

Attribute Recognition on Novel Images: The goal of this task is to label a given image with a single attribute name. Although there may be multiple attributes in a single image, for being able to compare our results on benchmark data-sets we consider one attribute label per image. For this purpose, first we divide the test images into grids in three levels using spatial pyramiding [32]. Non-overlapping patches (with the same size of training patches) are extracted from each grid of all three levels. Recall that, we have multiple classifiers for each attribute trained on different salient clusters. We run all the classifiers on each grid for all patches. Then, we have a vector of confidence values for each patch, corresponding to each particular cluster classifier. We sum those confidence vectors of each patch in the same grid. Each grid at each level is labelled by the maximum confidence classifier among all the outputs for the patches. All of those confidence values are then merged with a weighted sum to a label for the entire image. $D^i = \sum_{l=1}^{3} \sum_{n=1}^{N_l} \frac{1}{2^{3-l}} h_i e^{-(\hat{x}-x)/2\sigma^2}$ Here, N_l is the grid number for level l and h_i is the confidence value for grid i. We include a Gaussian filter, where $\hat{x}$ is center of the image and x is location of the spatial pyramid grid, to give more priority to the detections around the center of the image for reducing noisy background effect.

For evaluation we use three different datasets. The first dataset is Bing Search Images curated by ourselves from the top 35 images returned with the same queries we used for initial images. This set includes 840 images in total for testing. Second dataset is Google Colour Images [53] previously used by [53] for learning colour attributes. Google Colour Images includes 100 images for each color name. We used the whole data-sets only for testing of our models learned on a possibly different set that we collected from Google, contrary to [53]. The last dataset is sample annotated images from ImageNet [47] for 25 attributes. To test the results on a human labelled dataset, we use Ebay dataset provided by [53] which has labels for the pixels in cropped regions. It includes 40 images for each colour name.

Figure 6 compares the overall accuracy of the proposed method (**CMAP**) with three other methods on the task of attribute learning. As the baseline (**BL**), we use all the images returned for the concept query to train a single model. As expected, the performance is very low suggesting that a single model trained by crude noisy web images performs poorly and the data should be organised to train at least some qualified models from coherent clusters in which representative images are grouped. As two other methods for clustering the data, we used k-means (**KM**) and original SOM algorithm (**SOM**) with optimal cluster number, decided by cross-validation of whole pipeline, and again train a model for each cluster. The low results support the need for pruning of the data through outlier elimination. Results show that, CMAP's clusters are able to detect coherent and clean representative data groups so we train less number of classifiers by eliminating outlier clusters but those classifiers better in quality and also, on novel test sets with images having different characteristics than the images used in training, CMAP can still perform very well on learning of attributes.

Our method is also utilised for retrieving images on EBAY dataset as in [53]. [53] learns the models from web images and apply the models to another set so both method study a similar problem. We utilise CMAP with patches obtained from the entire images (**CMAP**) as well as from the masks provided by [53] (**CMAP-M**). As shown in Figure6 Right, even without masks CMAP is comparable to the performance of the PLSA based method of [53], and with the same setting CMAP outperforms the PLSA based method with significant performance difference.

On ImageNet dataset, we obtained 37.4% accuracy compared to 36.8% of Russakovsky and Fei-Fei[47]. It is also significant that, our models trained from different source of information are better to generalized for some of worse performance classes (rough, spotted, striped, wood) of [47]. Recall that we globally learn the attribute models from web images, not from any partition of the ImageNet. Thus, it is encouraging to observe better results in such a large data-set against [47]'s attribute models trained by a sufficiently large training subset.

Attribute Based Scene Recognition: While the results on different datasets support the ability of our approach to be generalised to different datasets, we also perform experiments to understand the effect of the learned attributes on a different task, namely for classification of scenes using entirely different collections. Experiments are performed on MIT-indoor [45], and Scene-15 [32] datasets. MIT-indoor has 67 different indoor scene with 15620 images with at least 100 images for each category and we use

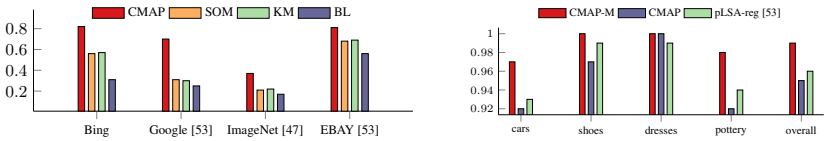


Fig. 6. Left: Attribute recognition performances on novel images compared to other methods. **Right:** Equal Error Rates on EBAY dataset for image retrieval using the configuration of [53]. CMAP does not utilise the image masks used in [53], while CMAP-M does.

100 images from each class to test our results. Scene-15 is composed by 15 different scene categories. We use 200 images from each category for our testing. MIT-indoor is extended and even harder version of Scene-15 with many additional categories.

We again get the confidence values for each grid in three levels of the spatial pyramid on the test images. However, rather than using a single value for the maximum classifier output, we keep the confidence values for all the classifiers for each grid. We concatenate these vectors for all grids in all levels to get a single feature vector of size $3xNxK$ for the image, which is then used for scene classification. Here N is the number of grids at each level, and K is the number of different concepts. Note that, while the attributes are learned in an unsupervised way, in this experiment scene classifiers are trained on the datasets provided (see next section for automatic scene concept learning).

As shown in Table1, our method for scene recognition with learned attributes (**CMAP-A**), performs competitively with [34] while using shorter feature vectors in relatively cheaper environment, and outperforms the others. Comparisons with [45] show that using the visual information acquired from attributes is more descriptive in the cluttered nature of MIT-indoor scenes. For instance, "bookstore" images has very similar structural layout to "clothing store" images, but they are more distinct with colour and texture information around the scene. Attribute level features do not create this much difference for Scene-15 data-set since images include some obvious statistical differences.

5.4 Learning Concepts for Scene Categories

As an alternative to recognising scenes through the learned attributes, we directly learn higher level concepts for scene categories. We call this method as **CMAP-S**. Specifically, we perform testing for scene classification for 15 scene categories on [32] and MIT-indoor [45] data-sets, but learn the scene concepts directly from the images collected from Web through querying for the names of the scene concepts used in these datasets. That is, we do not use any manually labelled training set (or training subset of the benchmark data-sets), but directly the crude web images which are pruned and organised by CMAP, in contrast to comparable fully supervised methods. As shown in Table1, our method is competitive with the state-of-the-art studies without requiring any supervised training.

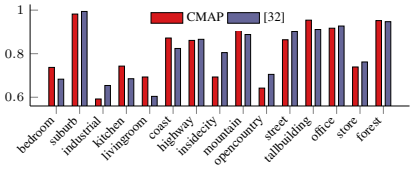
We then made a slight change on our original CMAP-S implementation by using the hard-negatives of previous iteration as a negative set of next iteration (we refer to this new method as **CMAP-S-HM**). We relax the memory needs with less but strong

Table 1. Comparison of our methods on scene recognition in relation to state-of-the-art studies on MIT-Indoor [45] and Scene-15 [32] datasets

-	CMAP-A	CMAP-S	CMAP-S+HM	Li et al. [34] VQ	Pandey et al. [44]	Kwitt et al. [30]	Lazebnik et al. [32]	Singh et al. [50]
MIT-indoor [45]	46.2%	40.8%	41.7%	47.6%	43.1%	44%	-	38%
Scene-15 [32]	82.7%	80.7%	81.3%	82.1%	-	82.3%	81%	77%

negative instances. As the results in Table1 and Figure7 show, we achieve better performances in Scene-15 than the state-of-the-art studies with this simple addition, still without requiring any supervisory input. However, on a harder MIT-indoor dataset, without using attribute information, low-level features are not very distinctive.

In order to understand the effect of discriminative visual features, which aim to capture representative and discriminative mid-level features, we also compare our method with the work of Singh et al. [50]. As seen in Table1, our performances are better than both their reported results on MIT-indoor, and our implementation on Scene-15.



	CMAP	[17]	[33]		CMAP	[17]	[33]
airplane	0.63	0.51	0.76	car	0.97	0.98	0.94
face	0.67	0.52	0.82	guitar	**0.89**	0.81	0.60
leopard	0.76	0.74	0.89	motorbike	**0.98**	0.98	0.67
watch	**0.55**	0.48	0.53	overall	**0.78**	0.72	0.75

Fig. 7. Left: Comparisons on Scene-15 dataset. Overall accuracy is 81.3% for CMAP-S+HM , versus 81% for [32] . Classes "industrial", "insidecity", "opencountry" results very noisy set of web images, hence trained models are not strong enough as might be observed from the chart. **Right:** Classification accuracies of our method in relation to [17] and [33].

5.5 Learning Concepts of Object Categories

We learn object concepts from Google web images used in [17] and compare our results with [17] and [33] (see Figure7 Right). [17] provides a data-set from Google with 7 classes and total 4088 gray scale images, 584 images in average for each class with many "junk" images in each class as they indicated. They test their results in a manually selected subset of Caltech Object data-set. Because of its raw nature of the Google images and adaptation to the Caltech subset, it is a good experimental ground for our pipeline.

Salient regions extracted from images are represented with 500 word quantized SIFT [36] vector with additional 256 dimension LBP [40] vector. In total we aggregated a 756 dimension vector representation for each salient region. At the final stage of learning with CMAP, we learn L2 norm, linear SVM classifiers for each cluster with negatives are gathered from other classes and the global outliers. For each learning iteration, we also apply hard mining to cull highest rank negative instances in the amount 10 times of salient instances in the cluster. All pipeline hyper-parameters are tuned via the validation set provided by [17]. Given a novel image, learned classifiers are passed over the image with gradually increasing scales, up to a point where the maximum class

confidences are stable. Among class confidences, maximum confidence indicates the final prediction for that image. We observe 6.3 salient clusters in average for all classes and 69.4 instances for each salient clusters. That is, CMAP eliminates 147 instances for each class as supposedly outlier instances. Results support that elimination of "junk" images gives significant improvements, especially for the noisy classes in [17].

5.6 Learning Faces

We use FAN-large [41] face data-set for testing our method in face recognition problem. We use Easy and Hard subsets with the names accommodating more than 100 images (to have fair testing results). Our models are trained over web images queried from Bing Image search engine for the same names. All the data preprocessing and the feature extraction flow follow the same line of [41], that is owned from [12]. However, [41] trains the models and evaluates the results at the same collection.

We retrieve the top 1000 images from Bing results. Face are detected and face with the highest confidence is extracted from each image to be fed into CMAP. Face instances are clustered and spurious face instances are pruned. Salient clusters are used for learning SVM models for each cluster in the same settings of the object categories. For our experiments we used two different face detectors. One is cascade classifier of [54] implemented in OpenCV library [4] and another is [57] with more precise detection results, even the OpenCV implementation is very fast relatively. Results are depicted at Table2 with two different face detection method and baseline result with models trained on raw Bing images for each person.

Table 2. Face learning results with detecting faces using OpenCV(CMAP-1) and [57](CMAP-2)

Method	GBC+CF(half)[41]	CMAP-1	CMAP-2	BaseLine
Easy	0.58	0.63	0.66	0.31
Hard	0.32	0.34	0.38	0.18

6 Conclusion

We propose Concept Maps for weakly supervised learning of visual concepts from large scale noisy web data. Multiple classifiers are built for each concept from clusters pruned from outliers, to have each classifier sensitive to a different visual variation. Our experiments show that we are able to capture low level attributes on novel images and have a good basis for higher level recognition tasks like scene recognition with inexpensive setting. We also show that we can directly learn scene concepts with the proposed framework. Going further, we show that CMAP is able to learn object and face categories from noisy web data. We are able to learn in an unsupervised way from the weakly-labeled web results and test on different datasets usually with different characteristics than the web data. Comparisons with the state-of-the-art studies in all tasks show that our method achieves better or similar results to the other methods which use the same/similar web data for training or which require supervision. As the future work, this framework will be extended to learn concepts from videos.

References

1. Berg, T.L., Berg, A.C.: Finding iconic images. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009, pp. 1–8. IEEE (2009)
2. Berg, T.L., Berg, A.C., Edwards, J., Maire, M., White, R., Teh, Y.W., Learned-Miller, E.G., Forsyth, D.A.: Names and faces in the news. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 848–854 (2004)
3. Berg, T.L., Forsyth, D.A.: Animals on the web. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1463–1470. IEEE (2006)
4. Bradski, G.: Dr. Dobb's Journal of Software Tools
5. Chen, X., Shrivastava, A., Gupta, A.: Neil: Extracting visual knowledge from web data. In: Proc. 14th International Conference on Computer Vision, vol. 3 (2013)
6. Choi, J., Rastegari, M., Farhadi, A., Davis, L.S.: Adding unlabeled samples to categories by learned attributes. In: CVPR (2013)
7. Couzinié-Devy, F., Sun, J., Alahari, K., Ponce, J.: Learning to estimate and remove non-uniform image blur. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1075–1082. IEEE (2013)
8. Doersch, C., Gupta, A., Efros, A.A.: Mid-level visual element discovery as discriminative mode seeking. In: Advances in Neural Information Processing Systems, pp. 494–502 (2013)
9. Doersch, C., Singh, S., Gupta, A., Sivic, J., Efros, A.A.: What makes paris look like paris? ACM Transactions on Graphics (TOG) 31(4), 101 (2012)
10. Endres, I., Shih, K.J., Jiaa, J., Hoiem, D.: Learning collections of part models for object recognition. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 939–946. IEEE (2013)
11. Erdem, E., Erdem, A.: Visual saliency estimation by nonlinearly integrating features using region covariances. Journal of Vision 13(4), 1–20 (2013)
12. Everingham, M., Sivic, J., Zisserman, A.: Hello! my name is... buffy–automatic naming of characters in tv video (2006)
13. Fan, J., Shen, Y., Zhou, N., Gao, Y.: Harvesting large-scale weakly-tagged image databases from the web. In: CVPR, pp. 802–809 (2010)
14. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. The Journal of Machine Learning Research 9, 1871–1874 (2008)
15. Farhadi, A., Endres, I., Hoiem, D.: Attribute-centric recognition for crosscategory generalization (2010)
16. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR (2009)
17. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google's image search. In: Tenth IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, pp. 1816–1823. IEEE (2005)
18. Ferrari, V., Zisserman, A.: Learning visual attributes. In: NIPS (2008)
19. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Automatic face naming with caption-based supervision. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
20. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Face recognition from caption-based supervision. International Journal of Computer Vision 96(1), 64–82 (2012)
21. Guillaumin, M., Verbeek, J., Schmid, C.: Multiple instance metric learning from automatically labeled bags of faces. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 634–647. Springer, Heidelberg (2010)

22. Harris, T.: A kohonen som based, machine health monitoring system which enables diagnosis of faults not seen in the training set. In: Neural Networks. IJCNN 1993, Nagoya (1993)
23. Hays, J., Efros, A.A.: Im2gps: estimating geographic information from a single image. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
24. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology 160(1), 106 (1962)
25. Jain, A., Gupta, A., Rodriguez, M., Davis, L.S.: Representing videos using mid-level discriminative patches. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2571–2578. IEEE (2013)
26. Juneja, M., Vedaldi, A., Jawahar, C., Zisserman, A.: Blocks that shout: Distinctive parts for scene classification. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 923–930. IEEE (2013)
27. Kim, G., Torralba, A.: Unsupervised detection of regions of interest using iterative link analysis. In: NIPS, vol. 1, pp. 2–4 (2009)
28. Kohonen, T.: Self-organizing maps. Springer (1997)
29. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: ICCV (2009)
30. Kwitt, R., Vasconcelos, N., Rasiwasia, N.: Scene recognition on the semantic manifold. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 359–372. Springer, Heidelberg (2012)
31. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 951–958. IEEE (2009)
32. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178. IEEE (2006)
33. Li, L.J., Fei-Fei, L.: Optimol: automatic online picture collection via incremental model learning. International Journal of Computer Vision 88(2), 147–168 (2010)
34. Li, Q., Wu, J., Tu, Z.: Harvesting mid-level visual concepts from large-scale internet images. In: CVPR (2013)
35. Li, Q., Wu, J., Tu, Z.: Harvesting mid-level visual concepts from large-scale internet images. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 851–858. IEEE (2013)
36. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
37. Marsland, S., Nehmzow, U., Shapiro, J.: A model of habituation applied to mobile robots (1999)
38. Marsland, S., Nehmzow, U., Shapiro, J.: Novelty Detection for Robot Neotaxis. In: Proceedings 2nd NC (2000)
39. Muñoz, A., Muruzábal, J.: Self-organizing maps for outlier detection. Neurocomputing 18(1), 33–60 (1998)
40. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 971–987 (2002)
41. Özcan, M., Luo, J., Ferrari, V., Caputo, B.: A large-scale database of images and captions for automatic face naming. In: BMVC, pp. 1–11 (2011)
42. Ozkan, D., Duygulu, P.: A graph based approach for naming faces in news photos. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1477–1482. IEEE (2006)

43. Ozkan, D., Duygulu, P.: Interesting faces: A graph-based approach for finding people in news. Pattern Recognition 43(5), 1717–1735 (2010)
44. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV (2011)
45. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR (2009)
46. Rastegari, M., Farhadi, A., Forsyth, D.: Attribute discovery via predictable discriminative binary codes. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 876–889. Springer, Heidelberg (2012)
47. Russakovsky, O., Fei-Fei, L.: Attribute learning in large-scale datasets. In: Kutulakos, K.N. (ed.) ECCV 2010 Workshops, Part I. LNCS, vol. 6553, pp. 1–14. Springer, Heidelberg (2012)
48. Schroff, F., Criminisi, A., Zisserman, A.: Harvesting image databases from the web. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(4), 754–766 (2011)
49. Simonyan, K., Parkhi, O.M., Vedaldi, A., Zisserman, A.: Fisher vector faces in the wild. In: Proc. BMVC, vol. 1, p. 7 (2013)
50. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 73–86. Springer, Heidelberg (2012)
51. Theofilou, D., Steuber, V., Schutter, E.D.: Novelty detection in a kohonen-like network with a long-term depression learning rule. Neurocomputing 52(54), 411–417 (2003)
52. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 776–789. Springer, Heidelberg (2010)
53. Van De Weijer, J., Schmid, C., Verbeek, J., Larlus, D.: Learning color names for real-world applications. IEEE Image Processing (2009)
54. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, p. I-511. IEEE (2001)
55. Yanai, K., Barnard, K.: Probabilistic web image gathering. In: Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval, pp. 57–64. ACM (2005)
56. Ypma, A., Ypma, E., Duin, R.P.: Novelty detection using self-organizing maps. In: Proc. of ICONIP 1997 (1997)
57. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2879–2886. IEEE (2012)