Revisiting Visual Pattern Mining

Tanmay Batra

CMU-RI-TR-17-22

May 2017

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Abhinav Gupta, Chair
Kris Kitani
Abhinav Shrivastava

*Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Robotics.*

Keywords: Visual Pattern Mining, Deep Learning, Fine Grained Recognition, Semi-supervised learning

Abstract

With the progress in deep learning based methods, visual pattern mining has seen a significant improvement in extracting visual patterns in the form of mid-level elements[18] and using these patterns for object recognition tasks. The problem with the previous approaches is that they are fully supervised and requires a large amount of labelled data for pattern mining. But how to make it work when there is little or no labelled data? In this work, we propose an unsupervised pattern mining algorithm which works very well given a large unlabelled dataset. We further extend it to show how it also adapts to include labelled data as well and thus, is able to extract information from both labelled and unlabelled data together. This property makes it very useful for low-shot recognition tasks where the labelled data is present in very small quantities and there is an abundance of unlabelled data. In this work we show the effectiveness of our pattern mining algorithm on the task of low-shot fine grained recognition and image labelling. We show that our unsupervised mining algorithm is able to detect fine grained patterns of good quality even without using any labels and if given a few labelled images there is a significant improvement in quality and diversity of patterns. We also show the ability of our approach in labelling more images from the large unlabelled pool and adding them iteratively to the labelled set in a semi-supervised learning based approach. Our method performs much better than the baselines which include previous state of the art approaches to fine grained recognition.

Acknowledgments

Firstly, I express my sincere gratitude to my advisor Prof. Abhinav Gupta for the continuous encouragement, guidance, and contribution of ideas for the development of this research. I thank my other thesis committee members Kris Kitani and Abhinav Shrivastava for being helpful and providing feedback on my work. I am thankful to Abhinav Shrivastava for collaborating on critical aspects of this work.

I am grateful to the members of our group for their input on my research as well as the presentation of my work. I would like to thank the rest of the RI community for many great memories. Finally, I thank my family members and friends for supporting my academic endeavors, I could not have done it without them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recently, a lot of work has been done in visual mid-level element discovery. Mid-level patterns are essentially patches in images which satisfy following two properties: Representativeness which means that the patterns should be present frequently enough and Discriminativeness which means that they should be present frequently in a particular category but not in rest of the world. These two properties make the patterns interesting as they can be effectively used to distinguish one object category from another. Figure 1.1 shows examples of such patterns for different animals. These mid level elements have been used a lot in tasks like classification, action recognition, discovering stylistic elements, etc.

A lot of techniques have been proposed in the last few years for extracting patterns. The traditional methods are based on hand crafted features and are less efficient. Very recently, Mid level Deep Pattern Mining [18] was the first work to propose an algorithm to mine for patterns in a fully automatic manner very efficiently. They also used CNN activations instead of HOG features (HOG is considered as a lossy descriptor) used by it's predecessors which resulted in a lot of improvement in their tasks. Another advantage of their approach is that it can handle large amounts of data really well and can mine patterns very efficiently irrespective of the number of images present. There are some issues with their approach. Firstly, the approach is fully supervised and thus, requires category labels for pattern mining process. It cannot be used with any unlabelled dataset. Their approach only works well when there is an availability of a large labelled dataset. But when we have only a few labelled images there is not enough information available to extract many diverse patterns that can generalize to the entire dataset which is another problem with their approach. In this work, we propose a way to mine patterns in an unsupervised way while ensuring diversity of detectors at the same time. Inspired by the techniques used in [18], we propose a visual pattern mining method that can work with a large pool of unlabelled data very efficiently and effectively. It requires no labels whatsoever. It is a fact that the availability of unlabelled data on the web is much more than labelled data and labelling data tends to be very expensive and inefficient. Our approach can take advantage of this and can easily utilize vast amount of information from the unlabelled data.

Although labelling a lot of data tends to be expensive, it is fairly inexpensive and easy to label data of the order of a few hundreds. In this work we further extend our pattern mining approach to handle this situation as well. Our approach can thus extract information from both labelled data and unlabelled data together and to our knowledge there has been no work previously in

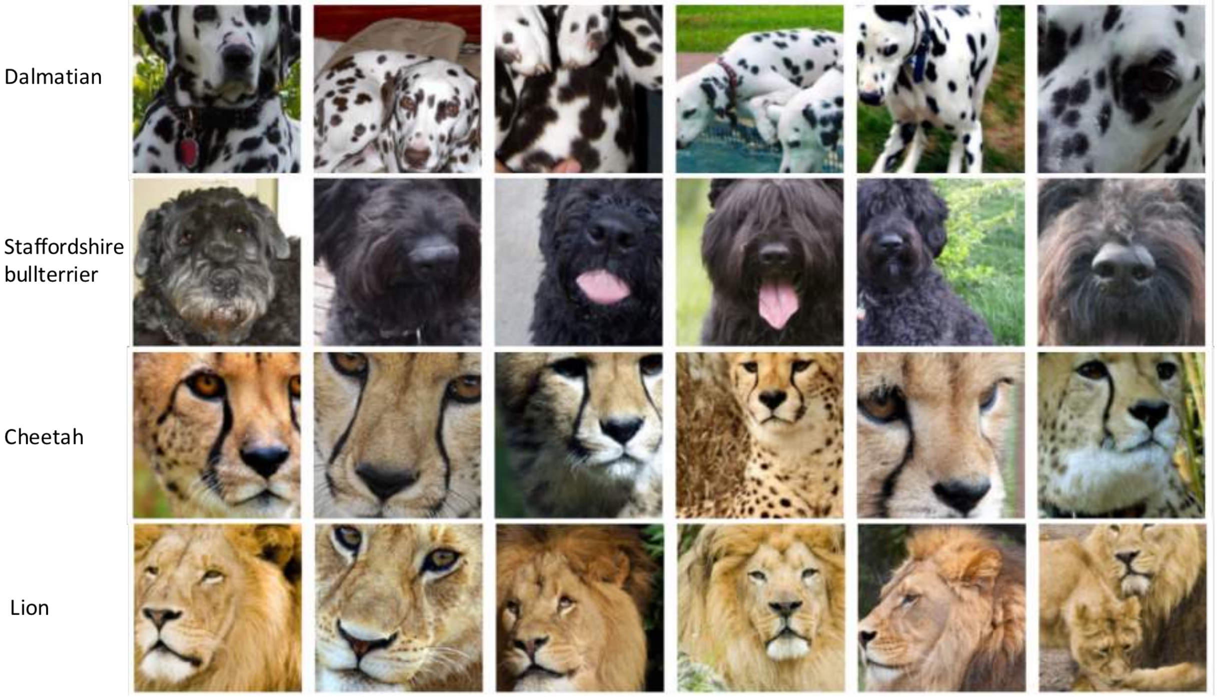

Figure 1.1: Examples of patterns for different animal categories.

this direction. It gives us the ability to use our approach for low-shot recognition tasks where usually there is an abundance of unlabelled data and a lack of labelled data. In this work we will focus on the task of low-shot fine grained recognition and we show that given very few labelled images, out approach performs much better that the fully supervised pattern mining approach in the previous work [18]. Since very few images are labelled the supervised algorithm proposed in [18] has very less information to mine patterns from as it can only use those few labelled images and their patterns may not generalize well to the entire dataset. In this work, we show that our method utilizes the large number of unlabelled images along with the few labelled images and are able to extract better patterns patterns and identify fine grained categories more accurately. Our patterns generalize much better to the entire dataset and has a much better coverage of categories of the dataset because the ability to use the large unlabelled dataset enables our algorithm to see and learn from all this extra information present and it tries to extract patterns from as many different images and as many different categories as it can.

Another contribution of our work is that we also use our approach to label the unlabelled data present very effectively and efficiently. We employ semi-supervised bootstrapping like approach to label more images and add those images to our labelled set. This further expands the utility of our approach and showcases how it can be used in so many different scenarios and applications. We show that our method outperforms all other previous baselines including the supervised mining method and other state-of-the-art approaches to fine grained recognition when starting with a small initial labelled set and a large unlabelled set. We show better performance in not only fine grained recognition task but the labelling of images is also much better when our approach is used compared to other baselines.

Chapter 2

Related Work

2.1 Mid level visual pattern mining

Mid level elements were first introduced by Singh et al. [23] and have been used extensively for tasks related to images such as image classification [6, 11, 17, 24], object detection[1], discovering stylistic elements[5, 16] and estimating geometry[13]. For videos, [10] have studied mid level pattern mining extensively provided examplar-SVM based procedure to mine for discriminative patches. But all these approaches are based on hand-crafterd features and typically have to search through thousand of image patches to find discriminative and representative patterns. As the size of data grows, this type of searching becomes more and more inefficient. Recently, [18] introduced a pattern mining approach for images integrated with deep learning and works well on large datasets. This approach handles big data very efficiently but a drawback of this approach is that it requires a large labelled data. It used information from all labelled images to extract interesting patterns and use them for recognition task. Our approach is inspired by their work and we propose a method which is capable of mining patterns in an unsupervised way and using all the untapped information in the large unlabelled data. In addition to that our method can also make use of a few labelled images as well to get even better and diverse patterns which enables our approach to be used in low-shot recognition tasks.

2.2 Fine Grained Recognition

A lot of work has been done in the area of fine grained recognition in the past. Many recent methods have taken advantage of part annotations , [2], [3], [29]; co-segmentation techniques [15] and bounding box annotations [20] for fine-grained recognition. On the other hand many traditional [7], [26], [27] and [28] as well as recent deep learning based methods [19] do not use additional annotations similar to our work.

But all of the above methods are fully supervised and use the entire labelled data in their approaches. How to deal with this when there is little to no data labels present ? Labelling a few images of the order of a few hundred is easy and inexpensive. We propose a low shot learning approach to tackle this problem. Given very few fine grained images, our approach is able to use both the labelled images and a large pool of unlabelled images to get patterns for all the fine grained categories and then recognize fine grained images using those patterns. In addition to low-shot recognition task, we employ semi-supervised bootstrapping based approach to further label the unlabelled set of images.

2.3 Semi-supervised Learning

Among the various levels of supervision that have been explored in the community, semi supervised learning approaches [30] try to achieve a good balance between maximizing accuracy while minimizing human input. A commonly used semi-supervised technique is the bootstrap or self-learning approach [30] where an agent initially learns from a small amount of labeled data. It then retrieves images from a large unlabeled pool whose labels it is most confident of, and transfers these to the labeled set to re-learn its model. Such approaches often suffer from semantic drift [4] and various approaches have been successfully used to resist this drift [4, 22]. In this work, our approach inspired from graph laplacian based approach [8] which has been shown to work very well for low-shot tasks.

Chapter 3

Approach

This section explains our overall approach. Each part of the approach is described in the following subsections. Section 3.1 describes the basics of a few data mining techniques, specifically frequent pattern mining and association rule mining .Section 3.2 describes the our unsupervised pattern mining algorithm in detail. Section 3.3 describes the motivation and process of adding a little supervision for low-shot tasks and modification of pattern mining algorithm. Section 3.4 describes the entire fine-grained recognition process in detail.

3.1 Data Mining: Market basket analysis

In this section, we will first explain what the core market basket method entails. Market basket analysis is a modelling technique used for data mining which models two things: how often a person buys a particular item from a store; and an "if-then" rule i.e. if he buys that particular item, what other items he tends to buy along with it.

We define an "itemset" $A = \{a_1, a_2, a_3, ..., a_n\}$ as the set of all *n* items present in the store. A "transaction" t_i is a subset of itemset *A* which refers to the items bought by person *i* in the store. We define a "transaction database" *D* $= \{t_1, t_2, t_3, ..., t_m\}$ as a set of transactions of *m* people who visited the store. For a subset *X* of the itemset *A* to be frequently occurring, we need to find the fraction of transactions containing *X*. This term is know as "support" of the itemset *X*.

$$supp(X) = \frac{\#\{t|t \in D, X \subseteq t\}}{m}$$

For any two subsets $X \subseteq A$ and $Y \subseteq A$, an association rule *X* $\rightarrow$ *Y* is an if-then rule implying that if items in *X* are bought then items in *Y* are also bought by the same person. "Confidence" of a rule *X* $\rightarrow$ *Y* is defined as fraction of people who buy items in both *X* and *Y* to the people who buy items in *X* only. This essentially illustrates a relationship between *X* and *Y*: how often *Y* is bought with *Y*.

$$conf(X \rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

Given the market basket approach, we next explain how we use this analysis for mining patterns in an unsupervised way first. After that we propose a way to incorporate the few labelled images we have available with us.

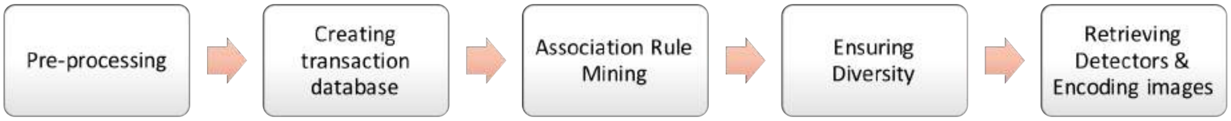
Pre-processing
Creating transaction database
Association Rule Mining
Ensuring Diversity
Retrieving Detectors & Encoding images

Figure 3.1: Unsupervised Pattern Mining process

3.2 Unsupervised Pattern Mining

In this section we will explain our pattern mining algorithm in detail. Our aim is to automatically find "interesting" patterns for each category. These patterns should represent that particular category. To identify such patterns, we follow two rules of pattern mining:

- **Representative**: For a pattern to be an interesting pattern, it should be representative of a category. It means that these patterns should appear frequently enough and present in majority of the videos in that category.
- **Discriminative**: A pattern is discriminative for a category if it distinguishes that particular category from others i.e. it is present frequently in the videos belonging to that category but not present frequently in other categories.

The entire process is given in figure 3.1. Each step in the process is explained below.

3.2.1 Pre-processing

In this section we will explain how we generate "patches" from images and extract their features which will be used for the pattern mining algorithm described below. From each image, we will first sample small patches at 2 scales: 128x128 and 64x64 pixels. The sampling stride is 32 pixels spatially. For each of the patches, we will extract it's 4096 dimensional CNN activation using the model we trained above. In this way we extract patches from all the images belonging to a dataset. The total number of patches extracted from each of the datasets amounts to almost 1 million on average. The next section explains how we discover a few "interesting" patterns from all the extracted patches.

3.2.2 Creating transaction database

Transactions must be created before any pattern mining algorithms can process. In our work, as we aim to discover patterns from image patches through pattern mining, an image patch is utilized to create one transaction. For creating transactions, we utilize two important properties of CNNs shown by [18] which are Sparsification and Binarization. As shown by them, the quality of CNN feature is fairly preserved when we perform binarization and sparsification depending upon the extent to which these properties are applied. To binarize a CNN vector, we only consider top 'k' values of the fc7 feature vector. In our experiments we set 'k' to be 30. The rest of the values are set to 0. After this, all non-zero values are set to 1. This creates a binarized vector from fc7. Now to sparsify it, we only consider indices of the non-zero values in the binarized

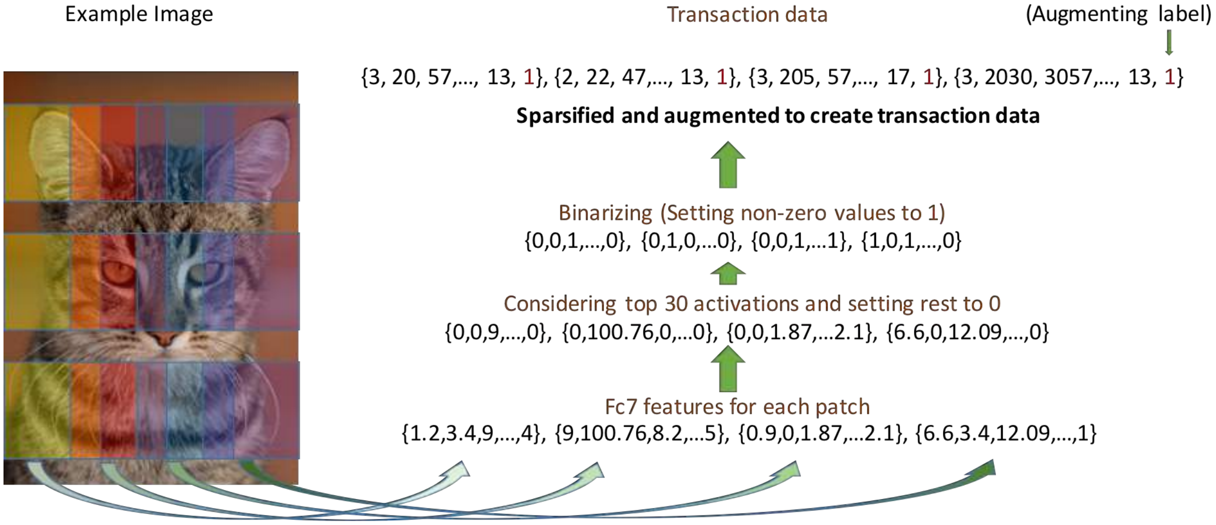

Figure 3.2: Transaction creation for an image

vector. Since there will be only 'k' indices, the fc7 vector has been converted to a 'k' dimensional vector.

Now we process on to creating the full transaction database. Firstly, we augment the fine grained dataset with a random set of images from imagenet (that do not belong to fine grained categories). In this case, the entire fine-grained dataset will serve as the positive data with +1 class label and the random images we added will serve as negative data with class label -1. Now we extract patches and the corresponding fc7 features from each image in this augmented dataset as described in section **??**. Now for each patch we have a 4096 dimensional feature vector and we will use the sparsified version of the vector as explained earlier. To this 'k' dimensional vector, we augment '+1' if the patch belongs to an image in our fine grained dataset. Otherwise, we augment '-1'. Thus, each transaction in our transaction database represents one patch the dimension of the transaction is 'k+1'. We make such transactions from all the patches collected from all the images in the augmented dataset. For every patch, we perform this step. Figure 3.2 shows this process for an image. Collection of transactions corresponding to all the patches creates the transaction database.

3.2.3 Association Rule Mining

Once we have a transaction database, we use Apriori algorithm to mine for a set of patterns using association rule mining. Apriori algorithm is essentially a fast way of counting and extracting relevant information form a large transaction database. Basically it follows the principle that if one element is not frequent then any combination of this element with any other element can never be frequent. For example, if an element '23' is found 5 times in the entire transaction database, a combination 23,4 cannot occur more than 5 times. Thus, the algorithm starts with single length elements and selects the subset which are frequent which gives all length 1 frequent elements. It then considers all combinations of length 2 from elements in this subset and finds out which combinations are frequent. This gives all frequent length 2 elements. The process

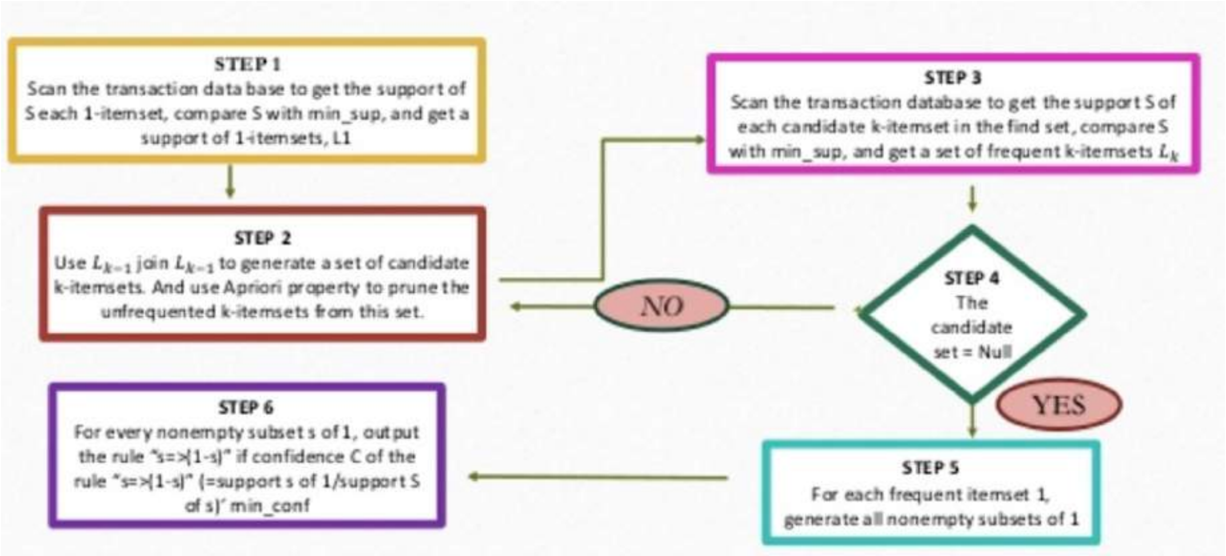
STEP 1
Scan the transaction data base to get the support of S each 1-itemset, compare S with min_sup, and get a support of 1-itemsets, L1
STEP 2
Use L_{k-1} join L_{k-1} to generate a set of candidate k-itemsets. And use Apriori property to prune the unfrequented k-itemsets from this set.
STEP 3
Scan the transaction database to get the support S of each candidate k-itemset in the find set, compare S with min_sup, and get a set of frequent k-itemsets L_k
STEP 4
The candidate set = Null
NO
YES
STEP 5
For each frequent itemset 1, generate all nonempty subsets of 1
STEP 6
For every nonempty subset s of 1, output the rule "s=>(1-s)" if confidence C of the rule "s=>(1-s)" (=support s of 1/support S of s)' min_conf

Figure 3.3: Apriori algorithm.

continues until all elements of all possible lengths are found which are frequent. Figure 3.3 gives an overview of the algorithm. We show below how each of the patterns extracted satisfies our requirements for being an 'interesting' pattern. Each pattern *P* extracted satisfies two constraints:

- $supp(P) >$ *MinSupport*
  for some fixed value of *MinSupport*.
  Since the support of the pattern *P* satisfy above constraint, this ensures that the "representativeness" criteria of pattern mining algorithm is satisfied because only those patterns will be considered which are present frequently in the dataset. The patterns which occur rarely will not satisfy the above criteria.
- $conf(P- > +1) >$ *MinConf*
  for some fixed value of *MinConf*.
  This criteria ensures that only those representative patterns will be considered further which are present more often in the fine grained dataset and NOT in the random image set we augmented to the dataset (which had the label -1 for all the images). The idea behind this is that similar types of background (example sky, water, etc) occurs in most of the fine grained images and using the random set of images against that removes the background patterns since only those patterns will satisfy this criteria that won't be present in random images often. Since the background is usually similar in most of natural images, this will help remove patterns corresponding to background images.

3.2.4 Ensuring Diversity

Using association rule mining as described above, we have ensured that most of the background patterns will be removed and fine grained patterns will be extracted. We can see that the problem with this approach is that it will simply give us patterns which occur most frequently in the entire dataset (since the entire dataset is assigned +1 category). So it is possible that the top patterns extracted in this way may only include a few of the fine grained categories and not represent all of them. To ensure the diversity of patterns, we employ a greedy reward based approach. We first extract 100,000 patterns using the above approach. We assign each image of the fine grained dataset a score/reward of 1. Every extracted pattern is composed of patches from some images. The score of a pattern is taken as the sum of scores of all images represented by this pattern. Whenever we extract an pattern with maximum score, we reduce the score of the images represented by that pattern by a fixed amount. This will result diverse patterns because there is a high chance that the pattern selected next represents images most of which have a score of 1 i.e. they haven't been represented by any pattern before. Running this algorithm in an iterative manner results in diverse patterns. In this way we can order all the 100,000 patterns by extracting 1 pattern in each run of the loop. The top 10,000 patterns extracted in this way tend to be quite diverse in nature and increase the amount of fine grained classes represented by them. The diversity of patterns and coverage of categories increases by a lot and the patterns become more useful and effective for recognition tasks because not these patterns can easily be generalized to the entire dataset.

3.2.5 Retrieving detectors and encoding images

After extracting top patterns using the method proposed above, we have to convert the patterns to detectors which can be used for object recognition task. Similar to [18], for each extracted pattern we consider all the patches that pattern is composed of and merge the patches together using LDA based merging method where we first combine the features of all the patches a pattern is composed of, subtract the merged value by mean of the dataset and divide it by the covariance of the dataset. For a pattern 'p', the detector 'd' is given according to the equation below.

$$d = (\sum_{\forall patch \supset p} feature_{patch} - mean_{data})/cov_{data}$$

This technique when used for the top 10,000 patterns mined above will give us a final set of 10,000 detectors.

After retrieving the detectors, we can use them to encode an image and generate a new feature representation for the new image. To accomplish that we fire each detector on the image in a sliding window fashion and take the max score per detector per region encoded in a 2-level (1 1 and 2 2) spatial pyramid. We concatenate these responses from all the top detectors to form the feature vector for the image. For our experiments, we consider an image at 3 scales and the final feature representation of an image is the outcome of max pooling on the features from all three scales. Every image can be represented in this new feature space and further recognition task can be performed in this new feature space instead of fc7.

3.3 Adding supervision

3.3.1 Motivation: Problem with No labelled data

The pattern mining algorithm explained above is completely unsupervised and requires no labels. But after we encode all the images in the new feature space, the classification task would again require labels. Also, in such a high dimensional data, simple clustering techniques like k-means clustering or spectral clustering in any feature space don't work well so we couldn't directly cluster the images this way.

But we have a set of extracted patterns along with the large unlabelled dataset. Can we use patterns to somehow cluster the unlabelled set to extract categories automatically ? For this purpose, we tried bi-clustering technique. Biclustering is a data mining technique which allows simultaneous clustering of the rows and columns of a matrix. Firstly, to generate such a matrix, we used both our extracted patterns unlabelled images in fc7 feature space. We generated a $num_{detectors} \times num_{images}$ where each cell i, j of the matrix corresponds to maximum score when detector i was fired on image j is sliding window fashion. We tried to form bi-clusters of this matrix.

In our experiments, we tried several bi-clustering algorithms namely, Cheng and Church, Bipartite Spectral Graph Partitioning, OPSM, Iterative Signature (ISA) , Spectral Biclustering, Information Theoretic Learning (ITL), xMOTIF, Plaid, FLOC, BiMax, Bayesian Biclustering, LAS, Qubic and Fabia [9] but unfortunately none of them seemed to work. The patterns and images couldn't be clustered together. Figure 3.4 shows an example of this approach. Top 4 clusters are shown on CUB dataset [25]. The yellow labels on the images show actual categories of the images in the CUB dataset. Clearly we can see that bi-clustering approach fails to give good clusters as each cluster has images belonging to many categories in the dataset. None of the clusters we found were pure in terms of image categories.

3.3.2 Labelling a few images

Since the images could not be clustered wither directly or using bi-clustering techniques, we proposed to label a few images of the order of 2 images per category. Labelling at this scale is very easy and inexpensive and we propose a semi-supervised bootstrapping based approach for the recognition task. In addition to that, this approach labelled the unlabelled images automatically as well. Now since we have a few labelled images with us, can we extend our pattern mining algorithm as well to use these additional images ? In the next section, we modify our pattern mining algorithm such that it uses this small labelled set along with the unlabelled set to extract even better and diverse patterns.

3.3.3 Modifying Pattern Mining algorithm

Figure 3.5 shows the modified pattern mining approach incorporating labelled data as well. We add an additional step in the process. In the pattern mining algorithm we described previously, we added a diversity approach to increase the coverage of patterns. After the reward based procedure in that approach, we are able to get better and a diverse set of patterns by simply considering the

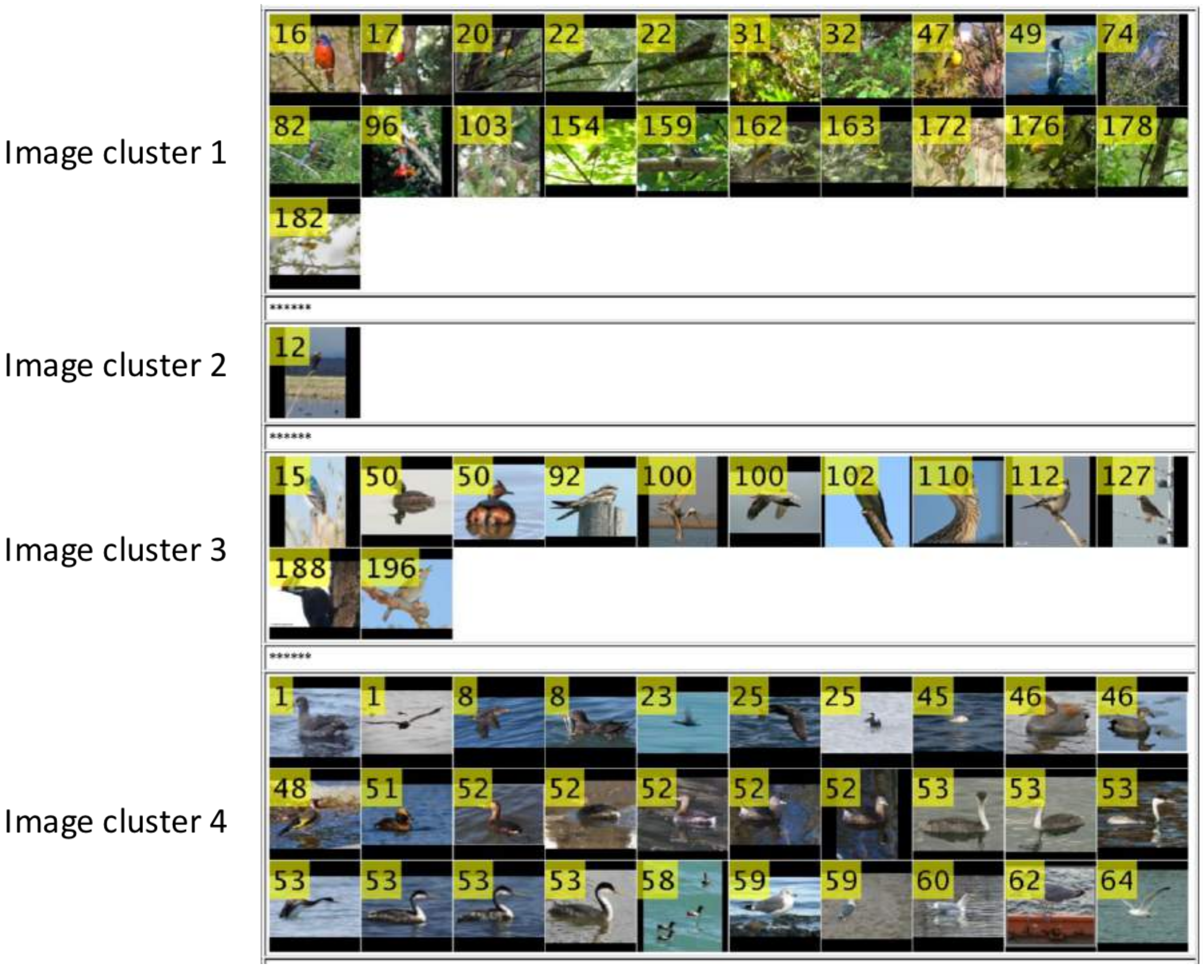

Figure 3.4: Failure case: Bi-clustering approach. Showing impure clusters formed.

top few patterns from the sorted order. But now we can make the pattern filtering procedure even better if we use a few labelled images for each fine grained category. For example, if we label 2 images per category we can use these 2 labelled images and get a subset from the set of patterns mined above which even higher diversity and perform better for fine-grained recognition. To accomplish this, we first take the patterns in the order which is the output of the previous step and again we use the similar reward based approach but this time we use only the few images we have labelled and we set score/reward on each category instead of each individual image. Thus, instead of considering what images are represented by each pattern, we will consider what categories are represented by each pattern and greedily select one pattern in each loop with the maximum score. This will give us a more diverse subset of patterns because it will ensure that the number of categories covered by the top most patterns in the new order cover as many classes as possible. In this way our proposed method has the ability to use the entire set of unlabelled images and the few labelled images to get a diverse set of patterns.

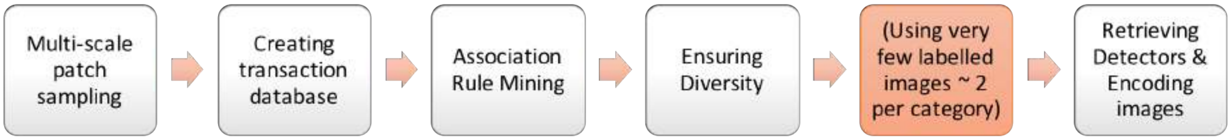
Multi-scale patch sampling
Creating transaction database
Association Rule Mining
Ensuring Diversity
(Using very few labelled images ~ 2 per category)
Retrieving Detectors & Encoding images

Figure 3.5: Modified pattern mining algorithm

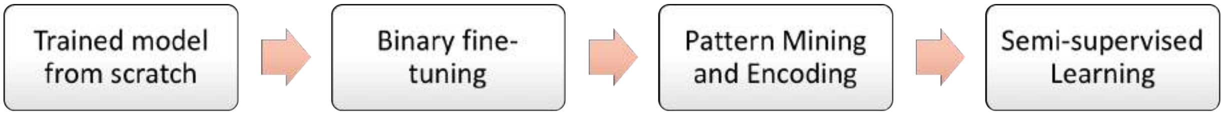

Figure 3.6: Low-shot Fine grained Recognition and Image Labelling.

3.4 Fine Grained Recognition and Image Labelling

We use our modified pattern mining algorithm for the task of low-shot fine grained recognition. We consider a fine grained dataset as a collection of a few labelled images (2 images per category). Rest of the images constitute the large unlabelled set. Figure 3.6 describes the entire process. Along with the recognition task, we emply semi-supervised learning based approach to label the unalelled dataset as well.

3.4.1 Training Networks from Scratch

To show the effectiveness of our approach, we use publically available fine grained recognition datasets which are frequently used in previous works on fine-grained recognition. We observed that there is some overlap between the test set of these fine grained datasets and the imagenet training dataset. Figure 3.7 shows examples of images which overlap between CUB test dataset and Imagenet training dataset. Thus, using a model pre-trained on imagenet would not be suitable in our case. So we train our base models from scratch. Since we are focussing on low-shot learning, we do not want our model to see even the training images of fine grained dataset before. In order to achieve this, we train one model for each dataset. For a particular dataset, we remove all images from the imagenet which have similar categories as the fine grained dataset we are using. For example, in case of Stanford Dogs dataset, we remove all 120 categories of dogs from imagenet and train our base model on remaining 880 categories. All our base models follow Caffe reference model architecture.

3.4.2 Binary Fine Tuning

While training our base models from scratch and removing all the categories corresponding to the fine-grained dataset, we ensured that the network hasn't seen any of those categories. But a deep network tends to perform better when it has an overview of what that object looks like. In order to incorporate this within our network, we do a binary finetuning of the network where the entire fine-grained dataset is treated as one category and the random negative dataset (an equal number of random set of images selected from the imagenet dataset) is treated as a second category and

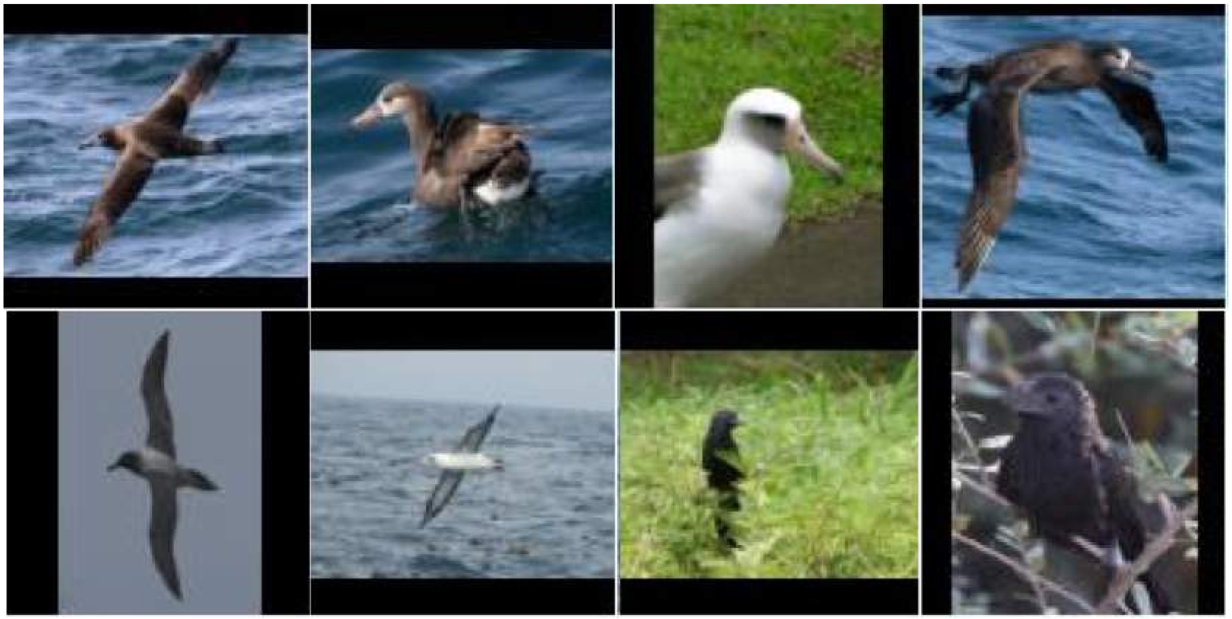

Figure 3.7: Some examples of test images from CUB dataset present in Imagenet training dataset

the network is fine tuned to classify fine-grained dataset from other random images. In this way for example we can make the network see what ”dogs” look like without using any fine-grained labels. As we show in our experiments, the binary fine-tuned network tends to perform better than the network trained from scratch even though we didn’t provide any fine-grained labels. This proves to be a good way to make the deep model learn about a particular object and it’s variations without explicitly providing labels for it.

3.4.3 Extracting patterns and encoding

After fine-tuning process, we use the images to extract patterns. We use our modified pattern mining approach which we described earlier for this purpose and incorporate both labelled and unlabelled data. After extracting patterns, we encode all the images in the new feature space as we explained earlier. The semi-supervised learning process occurs in this new feature space.

3.4.4 Semi Supervised learning

Given few labelled and many unlabelled images, we employ self learning bootstrap approach. We start with the initial labelled set and train predictors on this set. We run our predictors on the unlabelled set and transfer images with maximum confidence of belonging to a particular category to the labelled set. As the iterations continue, more and more images will be labelled and the labelled set will grow. There is one problem with using standard bootstrap method - semantic drift. Since we have very few images initially, the predictors trained are not very powerful. Thus they may assign wrong categories to unlabelled images with high confidence and those images will be transferred to the labelled set. Now the labelled set won’t remain pure since images are being transferred to incorrect categories. Now when we train predictors in the second iteration, they won't be trained correctly because the images they are trained on may not belong to correct categories. This error gets accumulated over iterations and more and more images are added to incorrect categories in the labelled set and the labelled set becomes more and more impure.

To solve this issue, we propose the use of 2 different predictors and only transfer images which have high scores corresponding to both of them. Such an approach resists semantic drift and image labelling becomes better. The 2 predictors we emply here are described below.

- **Graph Laplacian based approach**: Given a few labelled images per category and many unlabelled images, we use similar approach given in [8] for propagating labels to unlabelled images. This approach is based on binary classification. To use it for the task of multi-class classification, we run this for each category separately where the labelled images of all other categories are treated as negative in the algorithm. This outputs a probability value of an image belonging to the selected positive category. After performing this for each category, we obtain the probabilities of every image belonging to each category. This process is done in the new encoded feature space. We vary the parameters of the algorithm to get better clusters in each step of our approach. Since our data is not extremely big, we do not use approximate eigen functions. We use the exact eigen values and this improves our results.
- **Support Vector Classification:** We use the labelled images as the training set in the new encoded feature space and train a binary svm for each category seperately. The binary svm is trained so that it becomes consistent with the graph laplacian based method above and it's easy to combine the scores. We then use the trained svm model to predict labels and get the probability distribution of categories for all the unlabelled images.

Each approach above will assign category labels to every unlabelled image and provide the probability distribution of that image belonging to each category. For each image, we use a weighted average of the two probability values calculated by the two methods described above to get the final score. Thus, the final probability of an image *I* belonging to a category *C* is:

$$p(C|I)_{final} = p(C|I)_{method1} + p(C|I)_{method2}$$

where *method1* is the graph laplacian based approach and *method2* is the SVM based approach described above.

After computing the final probabilities of all the images, category of image *I* is assigned as

$$y = \arg\min_{C} p(C|I)_{final}$$

In our case, we transfer one image in each category which has the highest probability of belonging to that category. We can then perform pattern mining process again using the new set of labelled images. Note that in our pattern mining approach, the category labels are used only in the last step where we filter detectors based on category reward method. Thus we do not have to spend time in mining process in each iteration. The expensive mining process is only done once in the beginning. In each iteration we simply re-rank the detectors based on reward system using new set of labelled images and use the top detectors selected to encode the images again.

This above process is performed iteratively and we add one in each iteration which has the maximum score in each category to our set of labelled images. For testing, we use the provided test set of the fine grained dataset. The test set remains fixed across iterations. At each iteration we simply train an svm predictor using the labelled set in the current iteration on the new encoded feature space and calculate the prediction accuracy on the given fixed test set.

Chapter 4

Experiments

4.1 Datasets

We show our experiments on 4 fine-grained datasets namely CUB (Birds) dataset [25], Stanford Dogs dataset [12], Stanford Cars dataset [14] and the FGVC-Aircraft dataset [21]. The CUB dataset has 6033 images in total comprising of 200 fine grained categories and 3000 images for training. The training set includes 15 images per category. The Stanford Dogs dataset has 20,580 images in total comprising of 120 categories. There are about 12000 images in the training set. For both CUB and Dogs dataset, there are bounding boxes annotations available but we don't use these additional annotations in our work. The Cars dataset contains 16,185 images of 196 classes of cars. The data is split into 8,144 training images and 8,041 testing images, where each class has been split roughly in a 50-50 split. The aircrafts dataset contains 10,200 images of aircraft, with 100 images for each of 102 different aircraft model variants.

4.2 Results

In this section we will show the effectiveness of our pattern mining approach for extracting interesting patterns and performing low-shot fine grained recognition tasks based on the experiments on several datasets. All the experiments are based on Caffe Reference deep architecture. In the beginning we first show some ablation studies where we demonstrate how each of our choices in our mining algorithm has increased the performance of our approach. Next we use the best version of our approach in the semi-supervised setting for fine grained recognition and image labelling and compare it against 3 baselines namely alexnet fc7 features, supervised mining algorithm [18] and current state of the art bilinear cnn models [19].

4.2.1 Ablation study

Quantitative results

Table 4.1 shows the classification accuracies for CUB dataset for two scenarios. The first column is when we use all labelled images and the second column is when we use only 5 labelled images. There is no bootstrapping performed here. It's just fine grained recognition in the new encoded feature space. In this table, we show accuracies corresponding to addition to each and every choice we have made in our pattern mining approach. UPM stands for the basic unsupervised pattern mining algorithm without fine tuning or diversity approaches. $UPM + BFT$ adds binary fine tuning on top of basic method. $Reward(Images)$ and $Reward(Categories)$ add reward based approaches on top of the basic approach for images and categories respectively. Note that reward based approach for categories requires the labelled data as well. Reward based approach for images requires only unlabelled data as explained in our approach. As we can see that in both the scenarios the performance improves with each choice that we add to the approach which shows that every choice is effective. We also compare our methods with 4 baselines: $FishervectorSift$ which uses fisher vectors on sift space, $FC7$ which is recognition accuracy in fc7 feature space, SPM which is the recent work on supervised pattern mining algorithm and $BCNN$ which is the current state-of-the-art for fine grained recognition for CUB dataset. We can see that as the amount of labelled data decreases, our approach outperforms all these baselines. Our approach turns out to be the best one for low-shot recognition tasks,

Qualitative results

We saw that each of our choices have increased the quantitative performance of our algorithm. But what about qualitative performance ? To show the effectiveness of our diversity based approach, we visualize the detectors as well. For each of the datasets, we show the top patterns extracted using our unsupervised pattern mining algorithm without ensuring diversity. These examples are themselves show that even just using our unsupervised approach gives really good and consistent patterns. But after we employ our diversity algorithm, both with images and categories, we see that patterns corresponding to more categories come into picture. We cover categories that weren't present before. This further strengthens our claim along the quantitative results that the choices in our algorithm are indeed making a difference and improving performance.

Figure 4.1 shows a few examples of detectors from CUB dataset with just unsupervised pattern mining technique without using diversity approach. After we apply binary rewards based approach, we find that more diverse classes are discovered and figure 4.2 shows some of these additional categories discovered. This ensured that our diversity based approach is indeed working well and new diverse categories are being discovered. Similarly figure 4.3 shows the patterns for Stanford Dogs dataset before the diversity approach and figure 4.4 shows newly discovered categories after the diversity based approach. Figure 4.5 shows the patterns for Stanford Cars dataset before the diversity approach and figure 4.6 shows newly discovered categories after the diversity based approach. Similarly for the aircrafts dataset, figure 4.7 and figure 4.8 show the visualizations before and after diversity algorithm is employed respectively. The consistency of improvement in all the datasets shows that our approach is generalizable and can work for any dataset.

Method	Full dataset	5 labelled images per category
Fisher vector SIFT [19]	0.16	0.09
FC7 [19]	0.24	0.13
SPM (MDPM) [18]	0.35	0.22
SPM + BFT (MDPM) [18]	0.37	0.24
BCNN + BFT [19]	0.38	0.25
UPM	0.28	0.23
UPM+BFT	0.32	0.26
UPM+BFT+Reward(Images)	0.34	0.27
UPM+BFT+Reward(Images)+Reward(Categories)	0.35	**0.28**

Table 4.1: Ablation study: CUB dataset Fine grained Recognition

4.2.2 Fine grained Recognition

In this section we compare the performance of our approach with the baselines in the semi-supervised setting. For all the datasets, we start with 2 images per category and keep on adding 1 image in each iteration to the labelled set. Figure 4.9 for the CUB dataset, figure 4.10 for the Stanford Dogs dataset, figure 4.11 for Stanford Cars dataset and figure 4.12 for the Aircrafts dataset show that our approach performs much better than the baselines. Our method outperforms the performance for all the datasets. The recognition accuracy is much better across the iterations and our method also resists the semantic drift much more than other baselines and thus, the labelling of images is much more accurate in our case. Again, the consistency of improvement in all the datasets shows that our approach is generalizable and can work for any dataset.

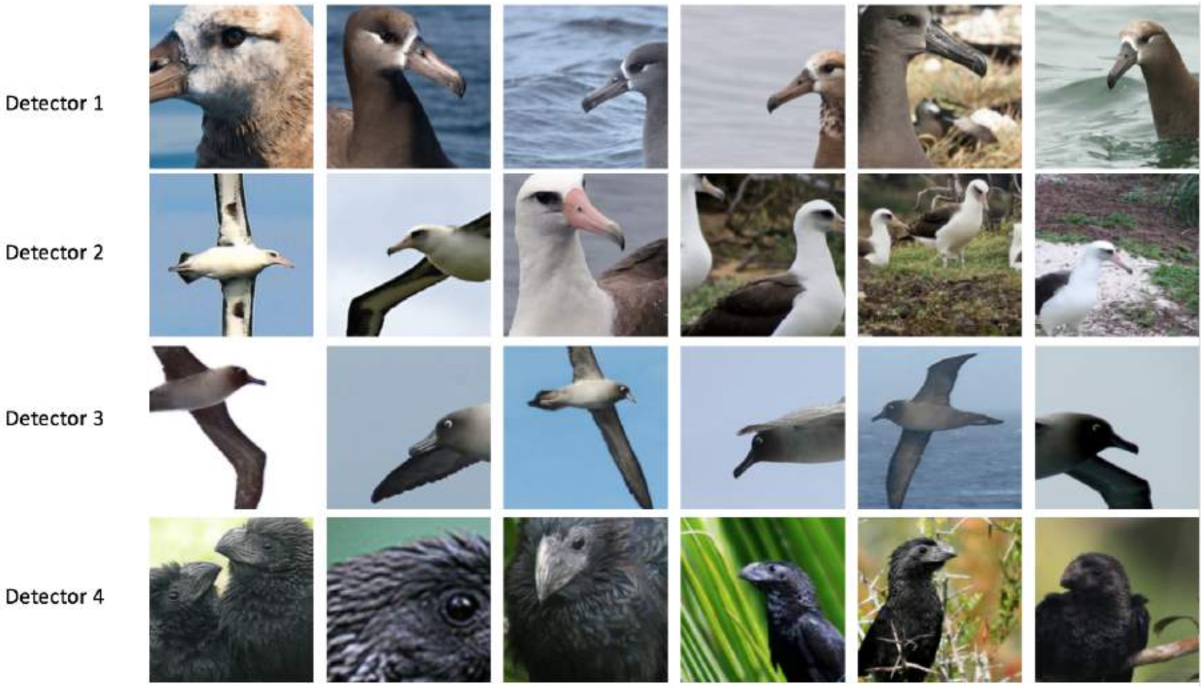

Figure 4.1: Unsupervised Patterns: CUB dataset

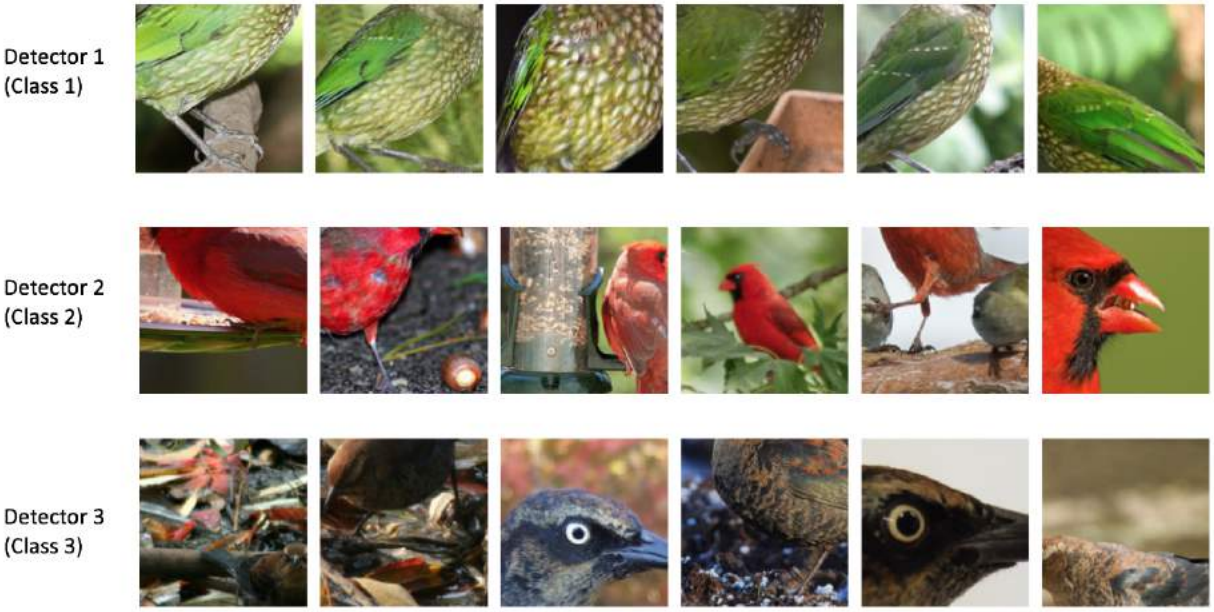

Figure 4.2: Addition of new classes and patterns with Reward based method: CUB dataset

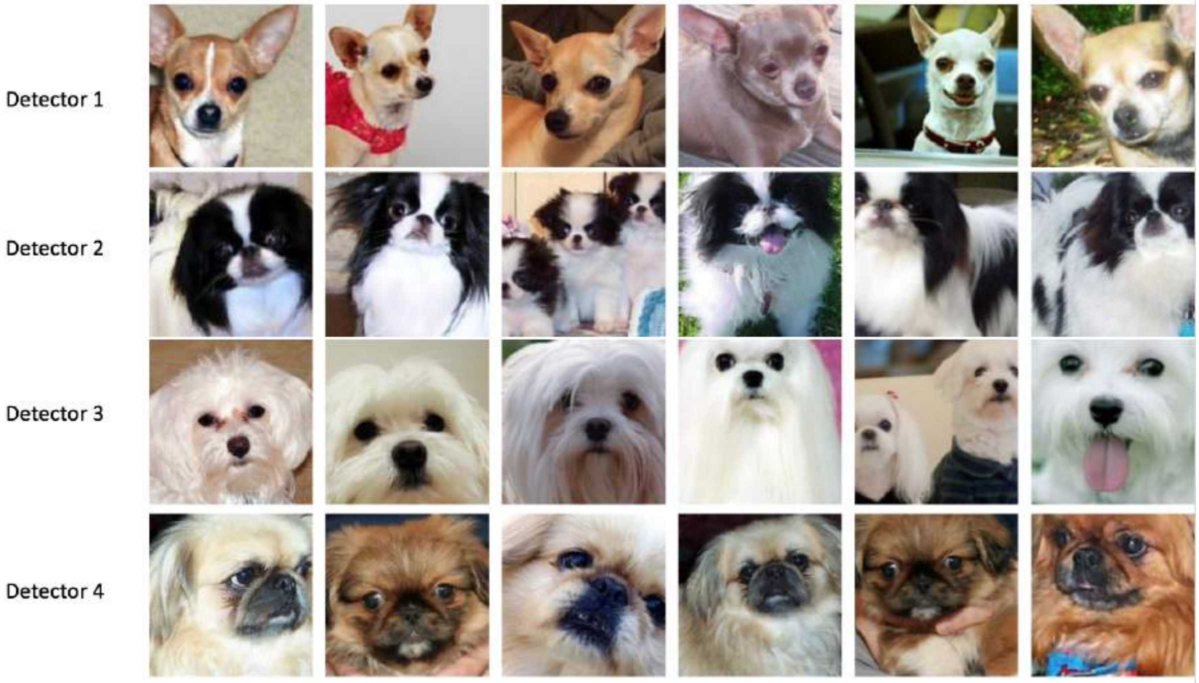

Figure 4.3: Unsupervised Patterns: Stanford Dogs dataset

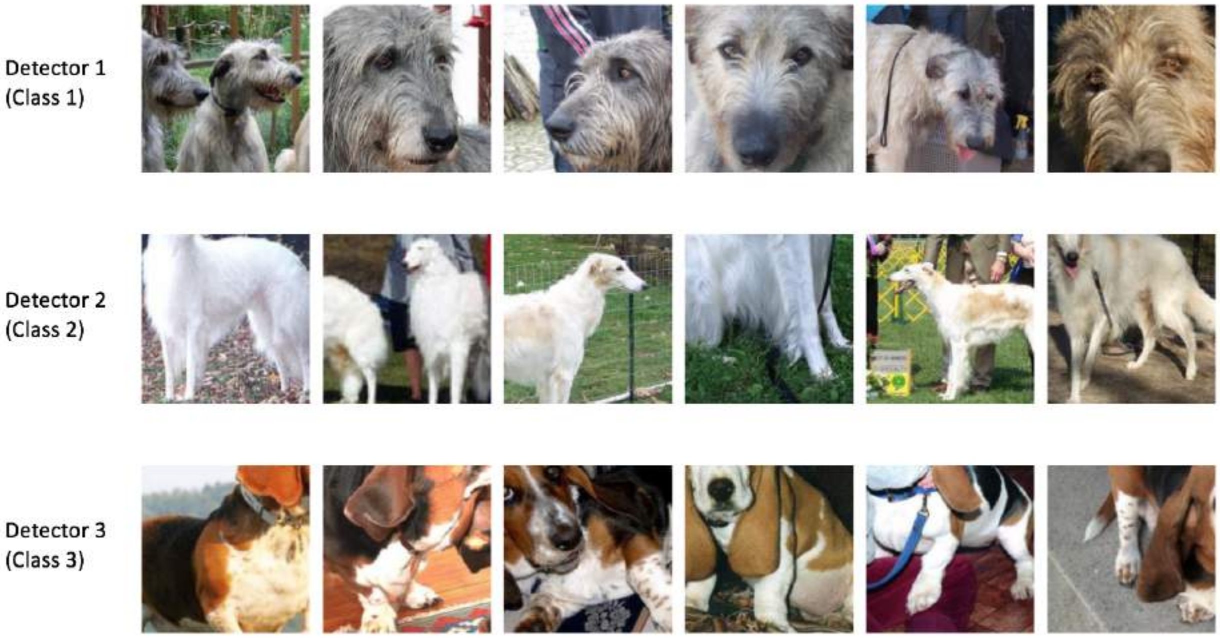

Figure 4.4: Addition of new classes and patterns Reward based method: Stanford Dogs dataset

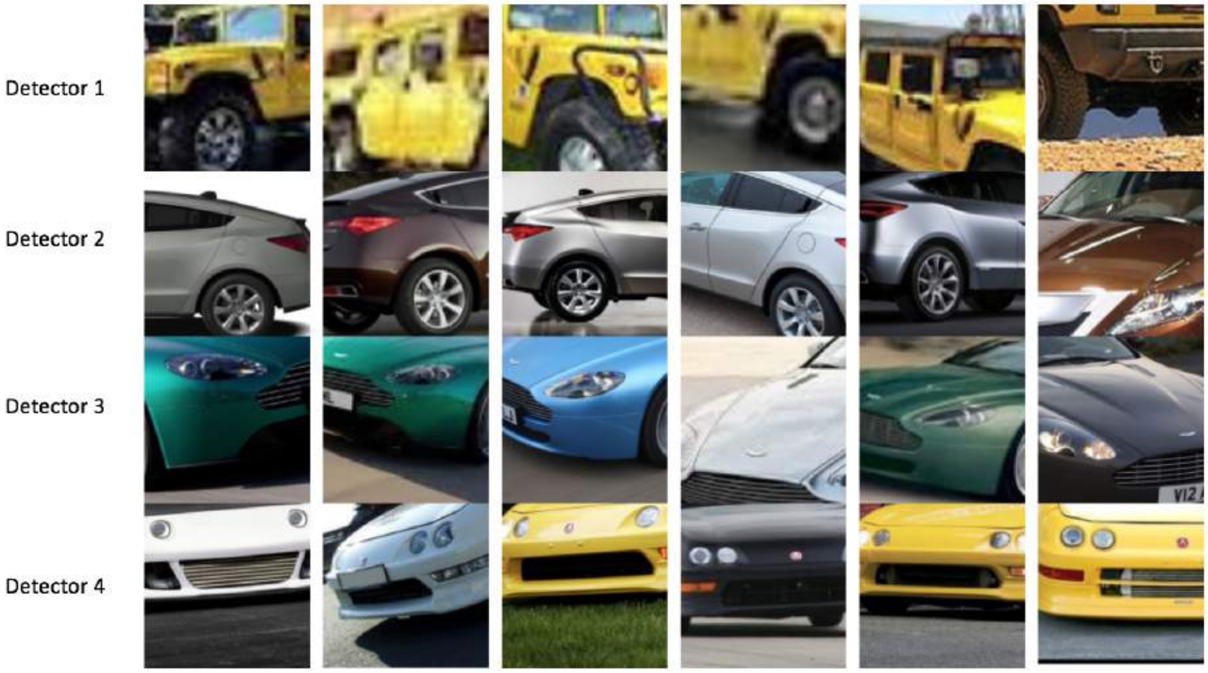

Figure 4.5: Unsupervised Patterns: Stanford Cars dataset

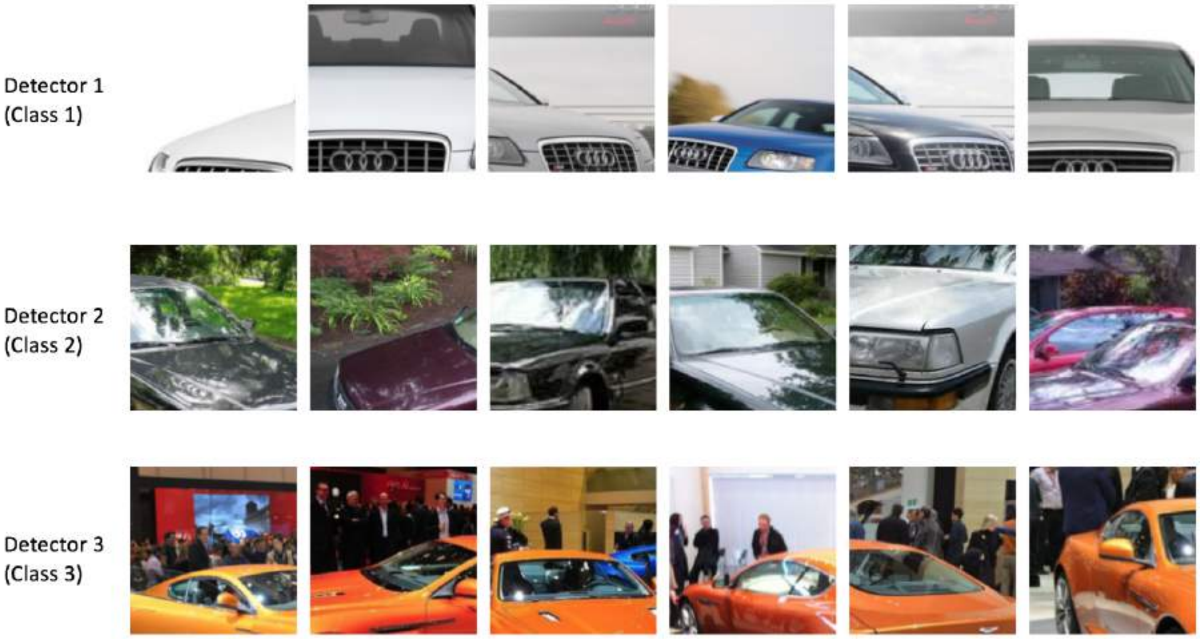

Figure 4.6: Addition of new classes and patterns Reward based method: Stanford Cars dataset

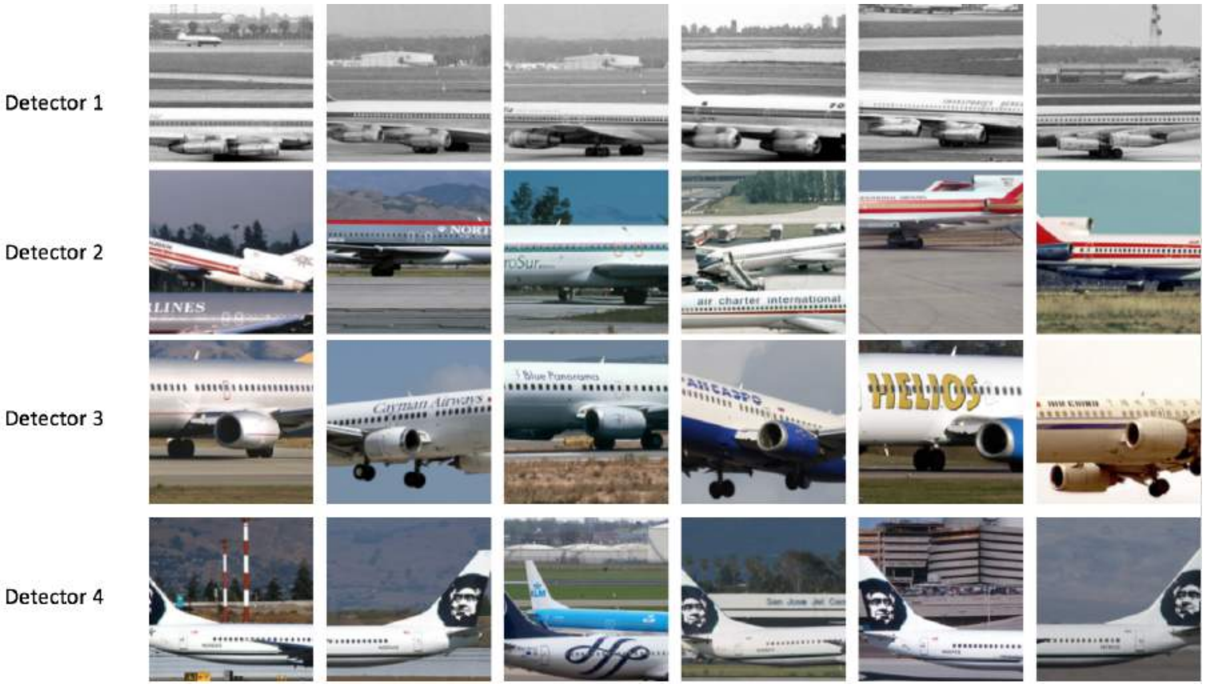

Figure 4.7: Unsupervised Patterns: Aircrafts dataset

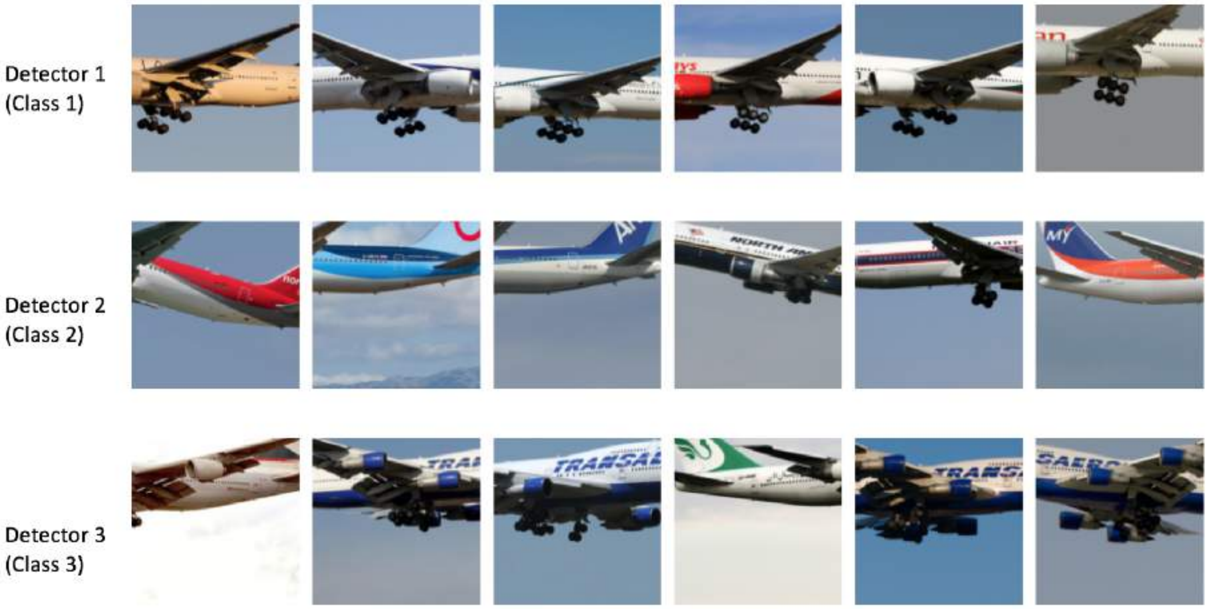

Figure 4.8: Addition of new classes and patterns Reward based method: Aircrafts dataset

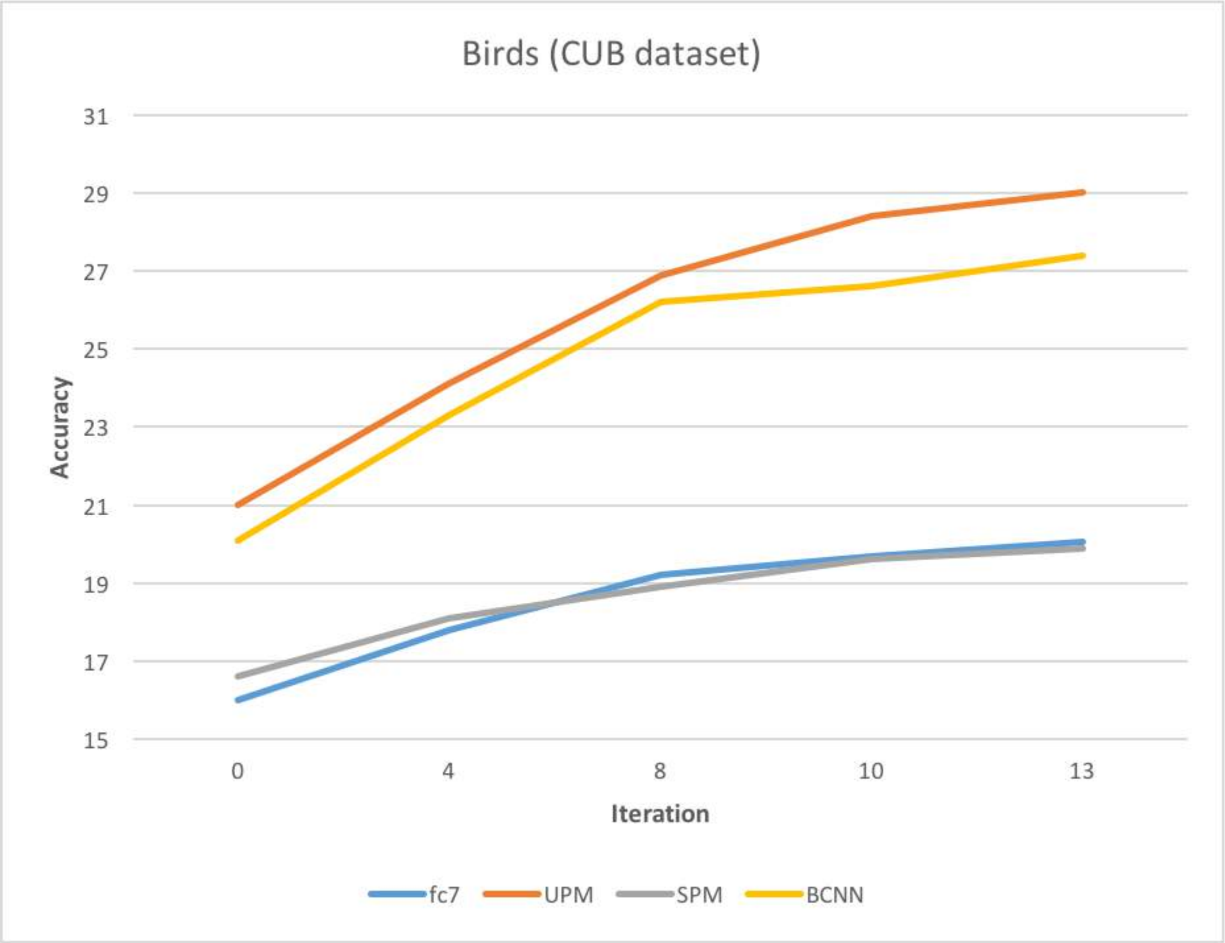

Figure 4.9: CUB dataset recognition accuracy across iterations

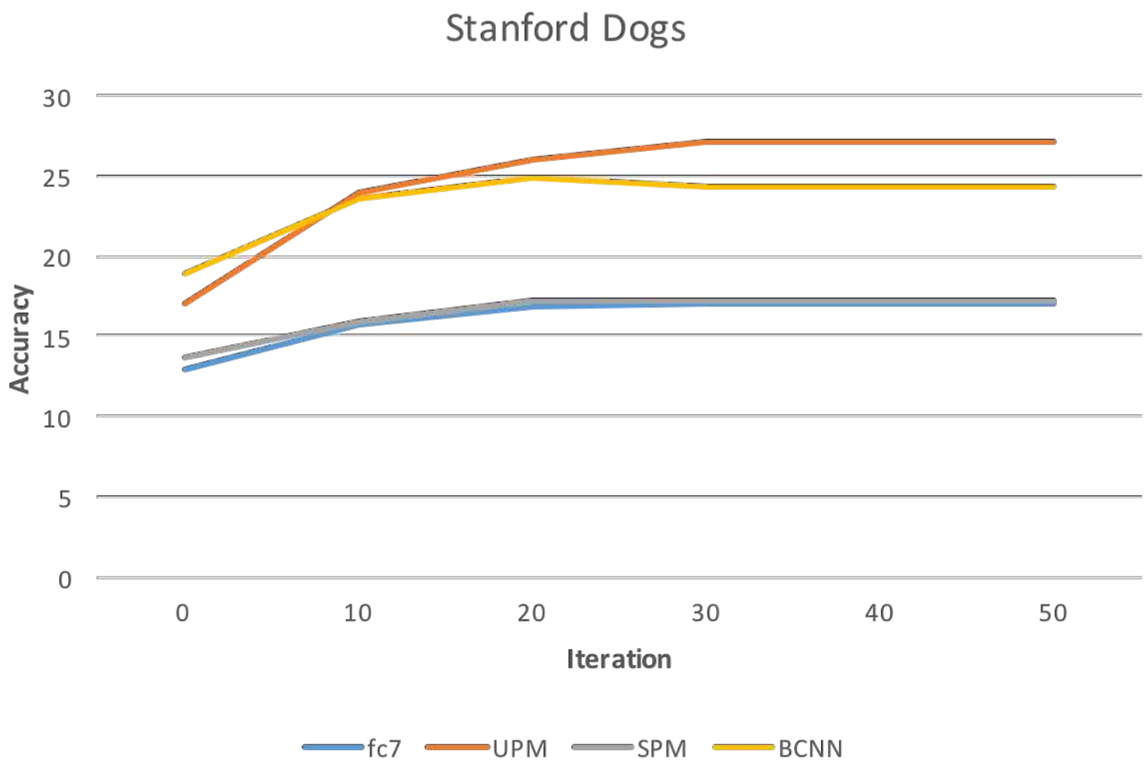

Figure 4.10: Stanford Dogs dataset recognition accuracy across iterations

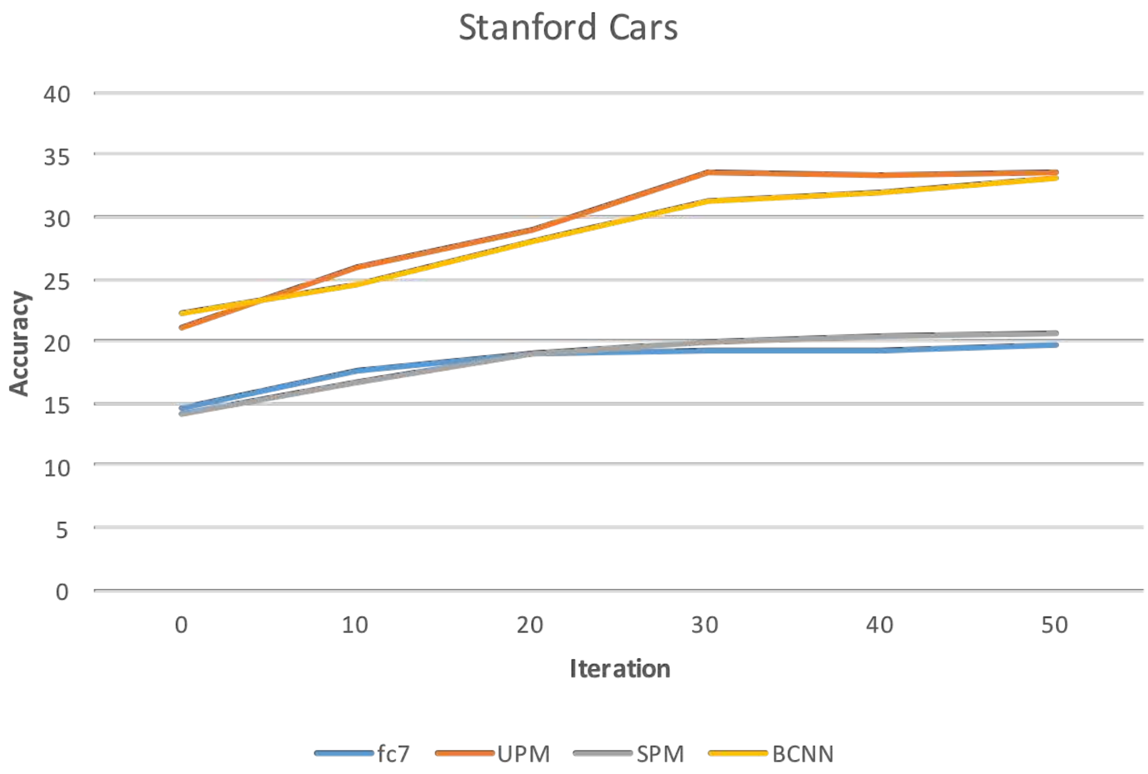

Figure 4.11: Stanford Cars dataset recognition accuracy across iterations

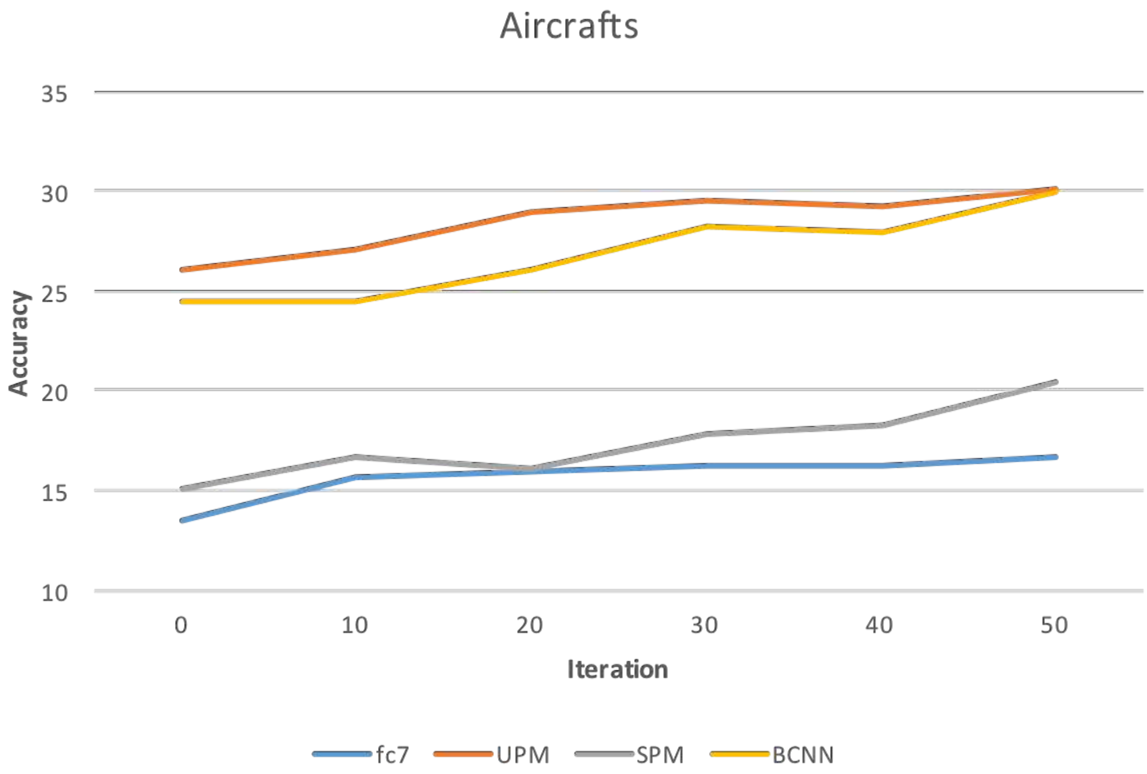

Figure 4.12: Aircrafts dataset recognition accuracy across iterations

Chapter 5

Conclusion

We have proposed an unsupervised pattern mining algorithm in this work and we show that it is an effective way of using untapped information from a large unlabelled data together with the few labelled images we have available with us and extract useful patterns which helps distinguishing among different fine grained categories. In this work we show the effectiveness of our pattern mining algorithm on the task of low-shot fine grained recognition and image labelling. We show that our unsupervised mining algorithm is able to detect fine grained patterns of good quality even without using any labels and if given a few labelled images there is a significant improvement in quality and diversity of patterns. We also show the ability of our approach in labelling more images from the large unlabelled pool and add them iteratively to the labelled set in a semi-supervised learning based approach. Our method performs much better than the baselines which include previous state of the art approaches to fine grained recognition.

Bibliography

[1] Aayush Bansal, Abhinav Shrivastava, Carl Doersch, and Abhinav Gupta. Mid-level elements for object detection. *arXiv preprint arXiv:1504.07284*, 2015. 2.1

[2] Thomas Berg and Peter N. Belhumeur. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, pages 955–962, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-4989-7. doi: 10.1109/CVPR.2013.128. URL `http://dx.doi.org/10.1109/CVPR.2013.128`. 2.2

[3] Steve Branson, Grant Van Horn, Pietro Perona, and Serge J. Belongie. Improved bird species recognition using pose normalized deep convolutional nets. In *British Machine Vision Conference, BMVC 2014, Nottingham, UK, September 1-5, 2014*, 2014. URL `http://www.bmva.org/bmvc/2014/papers/paper071/index.html`. 2.2

[4] James R. Curran, Tara Murphy, and Bernhard Scholz. Minimising semantic drift with mutual exclusion bootstrapping. In *In Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 172–180, 2007. 2.3

[5] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012. 2.1

[6] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Mid-level visual element discovery as discriminative mode seeking. In *Advances in neural information processing systems*, pages 494–502, 2013. 2.1

[7] Kun Duan, Devi Parikh, David Crandall, and Kristen Grauman. Discovering localized attributes for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 2.2

[8] Rob Fergus, Yair Weiss, and Antonio Torralba. Semi-supervised learning in gigantic image collections. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 522–530. Curran Associates, Inc., 2009. URL `http://papers.nips.cc/paper/3633-semi-supervised-learning-in-gigantic-image-collections.pdf`. 2.3, 3.4.4

[9] N. K. Verma J. K. Gupta, S. Singh. Mtba: Matlab toolbox for biclustering analysis. pages 94–97. IEEE, 2013. 3.3.1

[10] Arpit Jain, Abhinav Gupta, Mikel Rodriguez, and Larry Davis. Representing videos using

mid-level discriminative patches. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2571–2578, 2013. 2.1

[11] Mayank Juneja, Andrea Vedaldi, CV Jawahar, and Andrew Zisserman. Blocks that shout: Distinctive parts for scene classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 923–930, 2013. 2.1

[12] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011. 4.1

[13] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from rgb-d videos. *The International Journal of Robotics Research*, 32(8):951–970, 2013. 2.1

[14] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13, ICCV workshop)*, November 2013. URL `http://ai.stanford.edu/~jkrause/cars/car_dataset.html`. 4.1

[15] Jonathan Krause, Hailin Jin, Jianchao Yang, and Li Fei-Fei. Fine-grained recognition without part annotations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2.2

[16] Yong Lee, Alexei Efros, and Martial Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1857–1864, 2013. 2.1

[17] Quannan Li, Jiajun Wu, and Zhuowen Tu. Harvesting mid-level visual concepts from large-scale internet images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 851–858, 2013. 2.1

[18] Yao Li, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel. Mid-level deep pattern mining. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 971–980. IEEE, 2015. (document), 1, 2.1, 3.2.2, 3.2.5, 4.2, 4.2.2

[19] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *International Conference on Computer Vision (ICCV)*, 2015. 2.2, 4.2, 4.2.2

[20] L. Liu, C. Shen, and A. van den Hengel. The treasure beneath convolutional layers: cross convolutional layer pooling for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, 2015. 2.2

[21] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 4.1

[22] Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Constrained semi-supervised learning using attributes and comparative attributes. In *European Conference on Computer Vision*, 2012. 2.3

[23] Saurabh Singh, Abhinav Gupta, and Alexei Efros. Unsupervised discovery of mid-level

discriminative patches. *Computer Vision–ECCV 2012*, pages 73–86, 2012. 2.1

[24] Jian Sun and Jean Ponce. Learning discriminative part detectors for image classification and cosegmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3400–3407, 2013. 2.1

[25] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010. 3.3.1, 4.1

[26] Shulin Yang, Liefeng Bo, Jue Wang, and Linda G. Shapiro. Unsupervised template learning for fine-grained object recognition. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Lon Bottou, and Kilian Q. Weinberger, editors, *NIPS*, pages 3131–3139, 2012. URL `http://dblp.uni-trier.de/db/conf/nips/nips2012.html#YangBWS12`. 2.2

[27] Bangpeng Yao. A codebook-free and annotation-free approach for fine-grained image categorization. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 3466–3473, Washington, DC, USA, 2012. IEEE Computer Society. ISBN 978-1-4673-1226-4. URL `http://dl.acm.org/citation.cfm?id=2354409.2355035`. 2.2

[28] Bangpeng Yao, A. Khosla, and Li Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 1577–1584, Washington, DC, USA, 2011. IEEE Computer Society. ISBN 978-1-4577-0394-2. doi: 10.1109/CVPR.2011.5995368. URL `http://dx.doi.org/10.1109/CVPR.2011.5995368`. 2.2

[29] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. *Part-Based R-CNNs for Fine-Grained Category Detection*, pages 834–849. Springer International Publishing, Cham, 2014. ISBN 978-3-319-10590-1. doi: 10.1007/978-3-319-10590-1_54. URL `http://dx.doi.org/10.1007/978-3-319-10590-1_54`. 2.2

[30] Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005. URL `http://pages.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf`. 2.3