Feature Set Consolidation for Object Representation by Parts

Piyush Yadav[1,2], Shamsuddin Ladha[2], Shailesh Deshpande[2], Edward Curry[1]

[1]*Lero-Irish Software Research Centre, National University of Ireland Galway*
Galway, Ireland
{piyush.yadav,edward.curry}@lero.ie

[2]*Tata Research Development and Desing Centre, TCS Innovation Labs*
Pune, India
{shamsuddin.ladha,shailesh.deshpande}@tcs.com

***Abstract*—Machine learning based applications that run on image datasets increasingly use local image feature descriptors. We can visualize images as objects and local features as parts. Typically, there are thousands of local features per image, resulting in an explosion of feature set size for already huge image datasets. In this paper, we present a feature set consolidation strategy based on two aspects: pruning of non-discriminatory features across different object types and association of matching features for the same type of objects. We showcase the effectiveness of our consolidation strategy by performing classification on a building dataset. Our method not only reduces storage space footprint (~5%) and classification runtime (~4%) but also increases classification accuracy (~2%).**

Keywords—Object representation by parts, local image descriptor, part pruning, SIFT, classification

I. INTRODUCTION

Traditionally, for vision-based machine learning problems, each object instance in the dataset has been cohesively represented as a single—possibly very high dimensional—vector. This approach could be attributed to the observation that certain machine learning problems are better handled in high dimensional spaces (e.g. Support Vector Machine (SVM) [1]). However, with the availability of discriminating local features (such as SIFT [2]), it has been possible to conjure objects as a composition of parts (or components). For example, given an image of an animal (object), we refer as parts all the extracted local features (by a feature extraction algorithm). Here we have assumed that each image has only one object. This notion of representation by parts is different than the cognitive idea of physical parts of the animal, such as head, body, etc.

While locally discriminative features may provide a richer description of objects, they pose several new challenges for object representation and learning,

- A fixed high dimensional vector representation of the object is not practical as the number and order of local features generally vary across objects.
- The number of parts that represent a single object is typically in the order of hundreds (e.g. SIFT features) thus increasing the space and time complexity to store and process these features.
- Both intra-class and inter-class variability increase significantly due to richer object representation.
- The dimensionality of individual features is generally low, which can be problematic for learning algorithms where high dimensionality is desired (e.g. SVM).

To address these challenges, several approaches are found in the literature. These approaches can be broadly classified into three types, i.e., approaches that reduce a) the number of features; b) the feature dimensions; and c) both the number and dimensions of features.

Common approaches such as object representation by top N features or feature clustering aim at reducing the number of features. Ledwich et al. [3] reduced the number of SIFT features used for indoor scene representation based on the observation that a majority of the detected keypoints do not match between images that share a common camera viewpoint. Montazer et al. [4] represented objects by clustering SIFT features of an object with k-means to solve content-based image retrieval (CBIR) problem. Methods such as PCA-SIFT [5] and feature quantization [7] reduce feature dimensionality. For example, [6] used PCA-SIFT and applied locality sensitive hashing (LSH) for fast object retrieval. Hare et al. [7] introduced a method to quantize SIFT features based on inverted intensity images and then performed clustering to create a codebook for image matching. This method reduces feature dimensionality and the size of the feature set.

A key aspect that has not been addressed, in these methods is the elimination of features that are non-discriminative across different objects. Although SIFT features are discriminative for a given image, there is a possibility that some of these features may lose their discriminatory powers in the presence of similar features from altogether different objects. In other words, the distance between some of the SIFT features belonging to different objects could be small enough to cause incorrect feature matching across object types. In this paper, we intend to bridge this gap in the existing methods by pruning non-discriminative SIFT features across objects. In addition, we improvise feature clustering by integrating concepts of associative memory from [8]. Our approach is a combination of non-discriminatory feature elimination and associative feature clustering is called Feature Set "*consolidation*".

978-1-7281-5606-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ISM46123.2019.00029

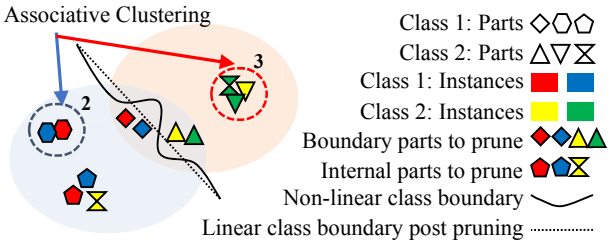

Figure 1: Object representation by parts

The approach can complement existing methods in the literature. In the following section, we describe in detail about feature set consolidation.

II. FEATURE SET CONSOLIDATION

Consider a simplified schematic, that demonstrates object representation by parts. Figure 1 shows two object instances of two different object classes. Each object instance is represented by three corresponding parts. Consolidation has two aspects, elimination and associative clustering. To explore the impact of consolidation in greater detail, we will study it in the context of the object classification problem.

Accurate classification of objects requires learning a complex decision boundary (in this case, non-linear). If we eliminate the non-discriminatory parts that lie near the decision boundary, then the boundary simplifies (becomes linear in this case). The elimination of non-discriminatory parts will not increase the complexity of the decision boundary because any non-discriminatory part could be labeled as internal (not impacting decision boundary) or boundary part.

In general, while building a classifier, it is important to strike a good balance between the errors due to bias and variance to have good generalization accuracy. It is evident that pruning of non-discriminative parts helps reduce not only bias—by reducing model complexity—but also variance—by reducing unwarranted variance in the dataset. Pruning, as demonstrated in the schematic (Figure 1), helps reduce errors due to bias and variance.

While the pruning of parts may sound similar to the pruning of object instances [9], there are some key differences. In general, pruning of parts—unlike object pruning—does not lead to loss of representation of the entire object(s) except when all the parts that compose object(s) are pruned. Object parts (across object classes) are likely to be closer to each other in the hyper-space than the corresponding objects due to the sheer number and low dimensionality of parts.

A large number of object parts allows us to not only be aggressive while pruning but also statistically determine the class of an object based on the underlying class distribution over parts. Thus, an object will be classified accurately as long as the majority of its parts are correctly classified.

Next, we look at the associative clustering aspect of feature set consolidation. Typical clustering methods substitute parts from same object class that belong to a single cluster with a single representative part. The cluster strength, i.e., the number of parts belonging to a cluster, which is indicative of the association of cluster members, is generally not utilized. In addition to typical clustering, our method stores the strength of each cluster. At the time of classification of an object, the cluster strength contributes to the class distribution that will be used to determine the class of the query object. Higher the cluster strength, larger will be the contribution to the distribution and vice-versa. This concept is inspired by sparse distributed memory (SDM) [8]. However, there are significant differences between SDM and associative clustering. While SDM permits clustering of instances—for the purpose of indexing—with same or different class labels, our approach only allows clustering of same label instances. Thus, clusters generated by our method are pure, i.e., all the cluster members belong to the same object class.

The two aspects of feature set consolidation, i.e., pruning and associative clustering are complementary. Together not only they contribute in reduction of storage space and classification runtime but also help strike a balance between model bias and dataset variance.

III. EXPERIMENTS AND RESULTS

To demonstrate the concept of pruning and its effectiveness, we have implemented a parts-based image classification system. The experiments consist of the following three steps:

1. Store the training dataset features in a database in an off-line mode with and without consolidation.
2. At runtime, match the test dataset features to those stored in the database. This results in the assignment of a class label to each feature in the set.
3. Derive a class label for each query object from its feature set label distribution (from Step 2).

Our experiments were conducted on a standard Intel Core i7 processor with Windows 10 operating system and 8GB RAM. The system was developed in Python with OpenCV port [11]. We have used the Zurich building dataset [10] for the experiments. The dataset with a total of 1005 images consists of five views of 201 different buildings in Zurich. Each view image has dimensions 640 x 480. Local features (SIFT) were extracted for each image where each SIFT feature interest point, represented a part of the image. The number of SIFT points extracted per image ranged from about 1000 to 2000.

The dataset was split into training and test sets with different percentages of data held out for testing. Two sets of classification experiments were conducted, one without and another with the consolidation of features. For the purpose of consolidation, at the time of training, the SIFT points were matched using a brute force matcher with $L2$ distance norm. Matched points with a distance less than a specific threshold were only considered as appropriate matches. This step of storing training data, without or with consolidation, was done in an off-line mode.

At runtime, during the classification of a query image, its local features were extracted and matched to the stored features resulting in distribution over class labels. The query image is

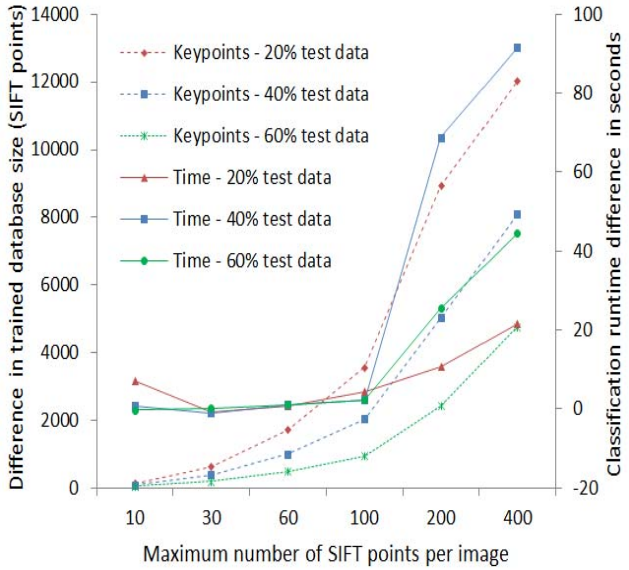

Figure 2: Experiments showcasing a reduction in the number of SIFT points stored and classification runtime due to pruning (Matching threshold = 100).

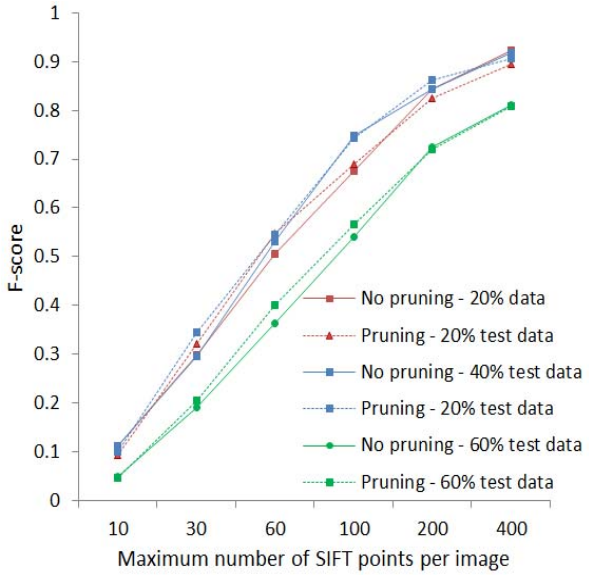

Figure 3: Graph with F-scores for different experiments with and without pruning (Matching threshold = 100).

labeled with the mode of the distribution. A range of experiments was conducted on the dataset to assess the impact of consolidation. Experiments with varying number of SIFT features, different size training datasets, and matching threshold was performed. Table 1, Table 2, and Table 3 respectively list the number of keypoints, classification time, and F-score values for these experiments. Figure 2 and Figure 3 respectively show the plot of difference, without and with consolidation, in the number of SIFT keypoints (left vertical axis) and classification runtime (right vertical axis), as a function of the maximum number of SIFT features extracted per image for three different sizes of test data. We see that as the number of extracted SIFT points per image increase, consolidation leads to a reduction in storage space and classification time. On average, our method reduces storage space requirements by 5% and shortens classification runtime by 4%. As the size of training set reduces, reduction in storage space decreases (the dashed lines). These results should be assessed in conjunction with the results of Figure 3 that show the classification performance of the two approaches (without and with consolidation) as the number of SIFT points extracted per image varied.

For each experiment, the assessment of the classification results was done by computing F-score. Table 3 lists the actual F-scores obtained for various experimental setups. Within a given experimental setup (e.g. with 20% dataset held out for testing), we observe that the F-score steadily increases for both without and with pruning experiments as the number of SIFT features increase.

The F-score for classification with consolidation is better on an average by about 2-3%, particularly for experiments where the number of SIFT features are limited (highlighted entries in Table 3). For other experiments, the F-scores with consolidation are similar to those without consolidation. This trend is observed for all the experimental setups (as seen in Figure 3) indicating no

Table 1: NUMBER OF FEATURES STORED FOR DIFFERENT EXPERIMENTAL SETUPS.

Dataset held out % →	Experimental Setup 1 (20%)		Experimental Setup 2 (40%)		Experimental Setup 3 (60%)	
SIFT features ↓ per image	Total features stored	Pruned features stored	Total features stored	Pruned features stored	Total features stored	Pruned features stored
10	8040	7892	6030	5946	4020	3974
30	24120	23488	18090	17722	12060	11875
60	48240	46515	36180	35183	24120	23632
100	80400	76837	60300	58255	40200	39249
200	160800	151861	120600	115565	80400	77981
400	321499	309461	241099	233006	160699	155940

Table 2: CLASSIFICATION RUNTIME (IN SECONDS) FOR DIFFERENT EXPERIMENTAL SETUPS.

Dataset held out % →	Experimental Setup 1 (20%)		Experimental Setup 2 (40%)		Experimental Setup 3 (60%)	
SIFT features ↓ per image	Time (No pruning)	Time (With pruning)	Time (No pruning)	Time (With pruning)	Time (No pruning)	Time (With pruning)
10	74.16	67.16	135	134.32	201.34	201.64
30	72.78	73.57	142.38	143.35	210.01	209.87
60	91.93	91.08	171.84	170.73	236.74	235.68
100	133.84	129.38	237.14	234.89	297.72	295.54
200	352.59	341.84	573.09	504.33	573.69	548.1
400	1007	985.38	1542.26	1450.8	1618.37	1574

adverse effect of the consolidation strategy on classification outcomes.A sample result of unrelated matching keypoints, one point from a building façade and another from a zebra crossing, which were pruned from two different building images, is shown in Figure 4. Figure 5 shows the impact of varying the keypoint

matching threshold. As the matching threshold increases, the number of points pruned increases significantly. However, the classification performance remains at par with that of without pruning experiments.

Table 3: CLASSIFICATION F-SCORE FOR DIFFERENT EXPERIMENTS.

Dataset held out % →	Experimental Setup 1 (20%)		Experimental Setup 2 (40%)		Experimental Setup 3 (60%)	
SIFT features ↓ per image	F-score (No pruning)	F-score (With pruning)	F-score (No pruning)	F-score (With pruning)	F-score (No pruning)	F-score (With pruning)
10	0.11	0.10	0.11	0.10	0.04	0.05
30	**0.30**	**0.32**	**0.30**	**0.34**	**0.19**	**0.20**
60	**0.51**	**0.55**	**0.53**	**0.54**	**0.36**	**0.40**
100	**0.68**	**0.69**	**0.75**	**0.74**	**0.54**	**0.57**
200	0.84	0.83	0.84	0.86	0.73	0.72
400	0.92	0.90	0.92	0.90	0.81	0.81

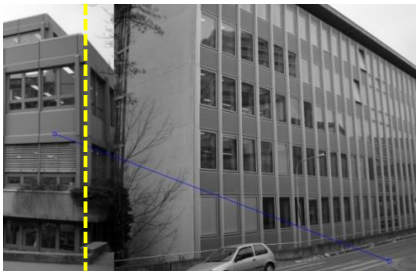

Figure 4: Sample matching points (in blue) from two different images (separated by the yellow marker) that were pruned.

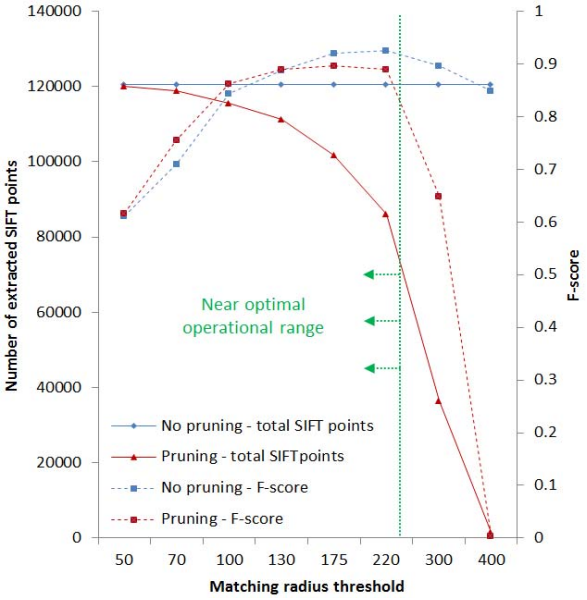

Figure 5: Impact of varying matching threshold on reduction in SIFT points and classification accuracy.

IV. SUMMARY

In this paper, we have presented a feature set consolidation strategy suitable for scenarios where objects (images) are represented by parts (local feature descriptors). During the process of consolidation, the non-discriminatory parts belonging to different objects—that often complicate machine learning models—were eliminated and the matching parts across different instances of the same object class were strengthened.

Our experiments demonstrate that such a strategy is efficient as it leads to savings in terms of both storage space and classification runtime without compromising classification performance. With increasing volume of image data and improved local descriptors, such consolidation strategies will be helpful in realizing increasing memory and time-intensive applications on limited computing power devices.

ACKNOWLEDGEMENT

This work was supported with the financial support of the Science Foundation Ireland grant 13/RC/2094 and co-funded under the European Regional Development Fund through the Southern & Eastern Regional Operational Programme to Lero - the Irish Software Research Centre (www.lero.ie). We also thanks to our collaborators at TCS Research for their active support in the work.

REFERENCES

[1] B. E. Boser, I. M. Guyon and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers," in Computational Learning Theory, New York, 1992.

[2] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

[3] L. Ledwich and S. Williams, "Reduced sift features for image retrieval and indoor localisation," in Australian Conference on Robotics and Automation, 2004.

[4] G. A. Montazer and D. Giveki, "Content based image retrieval system using clustered scale invariant feature transforms," International Journal for Light and Electron Optics, vol. 126, no. 18, pp. 1695-1699, 2015.

[5] K. Yan and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors," in Computer Vision and Pattern Recognition, 2004.

[6] J. J. Foo and R. Sinha, "Pruning SIFT for Scalable Near-duplicate Image Matching," in Conference on Australasian Database, Ballarat, 2007.

[7] J. S. Hare, S. Samangooei and P. H. Lewis, "Efficient Clustering and Quantisation of SIFT Features: Exploiting Characteristics of the SIFT Descriptor and Interest Region Detectors Under Image Inversion," in ACM International Conference on Multimedia Retrieval, 2011.

[8] P. Kanerva, Sparse Distributed Memory, Cambridge: MIT Press,1988.

[9] D. R. Wilson and T. R. Martinez, "Instance Pruning Techniques," in International Conference on Machine Learning, San Francisco, 1997.

[10] C. V. L. E. Zurich, "Zurich Building Image Database," [Online]. Available:http://www.vision.ee.ethz.ch/showroom/zubud/. [Accessed 02 May 2019].

[11] OpenCV Dev Team, "OpenCV-Python Tutorials," [Online]. Available: http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html. [Accessed 02 May 2019].

[12] J. Snaider, R. McCall and S. Franklin, "The LIDA Framework As General Tool for AGI," in International Conference on Artificial General Intelligence, Mountain View, 2011.