Fine-grained Collaborative K-Means Clustering

Tapabrata Chakraborti
Dept. of Computer Science
University of Otago
Dunedin, New Zealand
tapabrata@cs.otago.ac.nz

Brendan McCane
Dept. of Computer Science
University of Otago
Dunedin, New Zealand
mccane@cs.otago.ac.nz

Steven Mills
Dept. of Computer Science
University of Otago
Dunedin, New Zealand
steven@cs.otago.ac.nz

Umapada Pal
CVPR Unit
Indian Statistical Institute
Kolkata, India
umapada@isical.ac.in

***Abstract*—We present a collaborative clustering algorithm of fine-grained images, where the subtle inter-class differences are represented using collaborative filters. Cluster centers are represented as optimal weighted collaboration of data points, and the optimal weight matrix is analytically obtained. This may be viewed as a generalization of the K-means clustering algorithm, where these weights would be unity. We also introduce a matrix inversion scheme that allows us to scale up collaborative representations by orders of magnitude and this allows us to apply the scheme when the number of data points is large. Collaborative clustering outperforms K-means and a few of its popular variants K-medians, K-modes and K-medoids. It also outperforms DBSCAN and its recent variation DSets-DBSCAN. We have chosen species recognition (bird and butterflies) as a representative fine-grained image categorization problem, though the proposed algorithm is a general approach applicable to other similar tasks. We also introduce a new benchmark fine-grained image dataset, that of Indian endemic butterfly species (Titli.v1), which is available through the corresponding author.**

***Index Terms*—K-means Clustering, DBSCAN, Collaborative Representation, Fine-grained Recognition.**

I. Introduction

Supervised deep learning based vision systems have achieved near human accuracy in recent years. However, these methods work best when a large amount of well labeled and annotated data is available and there are many applications of practical significance where these prerequisites are not adequately met [1]. Automated recognition of endangered species in the wild [2] and detection of a rare pathology from medical images are examples of problems that may be characterized by scarcity of training images and imbalance of classes. Furthermore, these are highly specialized applications requiring labeling/annotations by domain experts. This might not be readily available and would be cost prohibitive to acquire in adequate quantity for deep learning. Thus unsupervised (weakly labeled or unlabeled) clustering methods are still pertinent for such applications.

It has been shown recently that collaborative filters are well suited for robust representation of fine-grained classes and give good results even on small datasets [3]. Collaborative filters have been the method of choice for recommender systems [4] and have recently also been used in vision systems like face recognition [5]. A collaborative representation classifier represents the query image as an optimal weighted average of the data samples across all classes and the final class label is assigned based on the least residual. In particular, collaborative representation classifiers have been shown to give good accuracy for fine-grained species recognition tasks [6], [7], which are challenging as shown in Figure 1. However, these methods have been mostly used for supervised recognition with available labels. Again considering the case of species recognition, there may be specific applications like identification of endangered species where there is a dearth of data labeled by domain experts. This suggests the need for an unsupervised clustering approach that can address these issues. The challenges of the fine-grained species recognition problem are illustrated in Figure 1.

In this work, we propose a collaborative k-means clustering algorithm. The collaborative cost function encodes the distances of each data point to the cluster centres and this function is optimised to find the representation weights. These weights are then used to update the cluster centers in each iteration. Thus, the proposed collaborative clustering may be viewed as a generalisation of the classic K-means algorithm [8]. K-means is a specialised case where the representation weights would be unity and hence the distance between data points would be Euclidean and the cluster centers would be updated by a simple mean. There are other methods referred to as "collaborative clustering" in existing literature, but these refer to a collaboration or ensemble of clustering methods [9], rather than using the collaborative filter analytically.

We compare the proposed collaborative clustering algorithm with K-means and several of its major variants: K-modes [10], K-medians [11] and K-medoids [12]. We also compare performance against DBSCAN [13], which is currently the most cited clustering method. We also compare against a state-of-the-art variation of DBSCAN called DSets-DBSCAN [14] where the authors present a non-parametric formulation based on dominant sets using similarity matrix of input data. We have chosen fine-grained species recognition as the representative problem. The tasks are bird species recognition (Ponce Birds [15] and Indian Birds [16] datasets) and butterfly species recognition (Ponce Butterflies [17] and the new Indian Butterflies Datasets). We take a dense variant of SIFT [18] as a feature descriptor. We also use an ensemble of GIST [19] and HoG [20] features as the second descriptor. It is seen that the proposed collaborative clustering easily outperforms K-means

978-1-7281-0125-5/18/$31.00

(a) (b) (c) (d)

Fig. 1: (a)-(b) are images of Kea and (c)-(d) are images of Kaka. These are NZ endemic birds, very similar in appearance. Of these, Kaka is endangered and Kea is vulnerable. Note subtle differences and pose variations (roosting vs. in flight).

and its variants, and also gives overall improvement against DBSCAN. The main contributions of this paper are:

1) **Collaborative Clustering**: We present a collaborative clustering algorithm for fine-grained data, as an optimal weighted generalization of the classic k-means.
2) **Indian Butterfly dataset**: We present a new fine-grained benchmark image dataset of Indian butterflies and report results on it. It currently has 60 images each of 6 types of butterflies.

II. Collaborative K-means Clustering

Collaborative filters represent the query sample as a weighted average of available data points across all categories of the dataset. The representation weights are then optimised via the collaborative cost function and the final categorization is assigned according to the sample with least residual. Collaborative filters should be well suited to represent fine-grained clusters with subtle differences and limited samples, since it finds optimal representation of data across clusters. The intuition is to incorporate this co-operative approach within the K-means clustering framework in this work. For clustering, this would translate to the cluster centers being represented as weighted mean of data points, where these weights are optimised via the collaborative cost function as analysed below.

Let the number of required clusters be K. Consider a dataset with N images in the feature space of d dimensions each, such that the feature matrix is $X \in \mathbb{R}^{d \times N}$. Choose K samples out of the N samples as a random initialisation of the cluster centres as $Y \in \mathbb{R}^{d \times K}$. Each cluster center is y_k, where $k = 1, \ldots, K$. α_k is the representation weight vector of dimension d for the cluster k.

The collaborative cost function is given by:

$$P(\alpha_k) = \|y_k - X\alpha_k\|_2^2 + \lambda\|\alpha_k\|_2^2 \tag{1}$$

The optimal value of α_k for each cluster center y_k are given by:

$$\hat{\alpha_k} = (X^T X + \lambda I)^{-1} X^T y_k \tag{2}$$

The residual for sample i with respect to y_k (k_{th} cluster) is given by:

$$r_i(y_k) = \frac{\|y_k - X_i\hat{\alpha}_{ki}\|_2^2}{\hat{\alpha}_{ki}^2} \tag{3}$$

Calculate $r_i(y_k) \forall i = 1, \ldots, N$ and $k = 1, \ldots, K$.

The sample i is allocated to the cluster center with lowest residual as follows:

$$C(X_i) = \arg\min_k r_i(y_k) \tag{4}$$

This concludes the first pass.

Let X^k be n_k number of columns of $X \in k_{th}$ cluster, $k = 1, \ldots, K$. $X^k = [X_1^k, \ldots, X_{n_k}^k] \in \mathbb{R}^{d \times n_k}$ where $\sum_{k=1}^{K} n_k = N$. Let $\hat{\alpha}_{kj}$ be the representation weight corresponding to $X_j^k \in X^k, k = 1, \ldots, n_k$.

In the next iteration, the new cluster centres are computed through:

$$y_k = \frac{1}{n_k} \sum_{j=1}^{n_k} X_j^k \hat{\alpha}_{jk} \tag{5}$$

Same steps are repeated until the termination condition is reached.

Reducing computation cost through SVD. The optimal representation weight matrix $\hat{\alpha}$ from eqn 2. has the term $(X^T X + \lambda I)^{-1}$, where X is of dimension $d \times N$. Here d is the dimension of the descriptor and N is the total number of data points in the dataset. This poses the problem of high computation cost for large datasets (N is large). So we use singular value decomposition (SVD) to reduce the matrix inverse computation to dimension $d \times d$, so as to make it independent of dataset size. This is a crucial modification needed for applications like image retrieval from large unlabeled or weakly labeled image repositories.

If we take the singular value decomposition (SVD) of X^T, we can factor $X^T X$ as:

$$X^T X = (USV^T)^T USV^T = VS^T U^T USV^T = V(S^2)V^T \tag{6}$$

Since S only has d non-zero singular values, we can truncate $S^T S$ and V to be smaller matrices. So V is $N \times d$, S is $d \times d$ and V^T is $d \times N$.

Using the Woodbury matrix inverse identity [21], the inverse term then becomes: $(VS^2V^T + \lambda I)^{-1}$

$$= \frac{1}{\lambda} + \frac{1}{\lambda^2} V(S^{-1} + \frac{1}{\lambda} V^T V)^{-1} V^T = \frac{1}{\lambda} + \frac{1}{\lambda^2} V(S^{-1} + \frac{1}{\lambda} I)^{-1} V^T \quad (7)$$

Note that the inverse term $(S^{-1} + \frac{1}{\lambda} I)^{-1}$ is only $d \times d$, so it will scale to many data points.

Algorithm 1: Collaborative K-Means Clustering

Choose number of clusters K **Initiate** the cluster centers randomly from the data points **Form** the feature matrix X and the cluster center matrix Y **Find** initial reconstruction vector α by eqn. 2. **while** *Termination condition is not reached* **do**
  for *each cluster center* $y_k \in Y$ **do**
    Find the collaborative weights α by eqn. 2.
    for *each image* $x \in X$ **do**
      Find the distances of x from cluster center y_k using eqn. 3 and 4.
    end
  end
  Update cluster centers by eqn. 5.
  Continue from Step 5 till termination condition reached.
end

III. Experimental Setup

In this section, we present the resource choices used for our experiments: the datasets, the feature descriptors, and the competing clustering methods for comparison.

A. Competing Clustering Methods

K-means and its variants. Collaborative clustering may be looked upon as a generalization of the K-means algorithm [8]. Collaborative clustering represents cluster centers as optimal weighted sums of data points. Thus, K-means (Lloyd's implementation [8]) is a specialized case where these weights are all unity and only the Euclidean distance from cluster centers is hence considered. We evaluate the performance of collaborative clustering against K-means and three of its major variants: K-modes [10], K-medians [11], K-medoids [12]. K-modes and K-medians, as the names suggest, utilise the cluster modes and medians instead of the means during the updates. The K-medoids algorithm chooses datapoints as centers (medoids or exemplars) and uses a generalization of the Manhattan Norm instead of the Euclidean distance.

DBSCAN. Density-based spatial clustering of applications with noise (DBSCAN) [13] is currently the most cited clustering algorithm. It groups together densely packed data points (with many nearby neighbours) and marks points in low density areas as outliers. Thus DBSCAN is somewhat robust to noise and unlike the K-means algorithms, does not require apriori knowledge of required number of cluster centers. We also compare against a recent state-of-the-art variation of DBSCAN, named D-Sets DBSCAN [14], where the authors present a non-parametric formulation based on dominant sets using similarity matrix of input data.

B. Benchmark Datasets

We test the proposed clustering algorithm on four species recognition datasets. One sample image from each class of the datasets are provided in Fig. 2.

- **Ponce Birds Dataset** has images of 6 bird species with 100 images per type [15]. It was compiled by the Ponce Research Group at the University of Illinois Urbana-Champaign. The 6 bird species are Egret, Mandarin duck, Snowy owl, Puffin, Toucan and Wood duck.
- **Indian Birds Dataset** was recently compiled at the Indian Statistical Institute in collaboration with the University of Otago, NZ [16]. It has 6 classes of endemic Indian birds, with 100 images per species. The six bird species considered are Black and Orange Flycatcher, Nilgiri Wood Pigeon, Nigiri Fly Catcher, Malabar Grey Hornbill, Forest Owlet and Rufous Babbler.
- **Ponce Butterfly Dataset** was compiled by the Ponce Research Group at the University of Illinois Urbana-Champaign [17]. It has 619 images of 7 classes of butterflies with images per class varying between 42 and 134. Out of these categories, 2 classes are of the same butterfly for wings closed and open (Monarch). We fuse these two classes. We take 40 images per class selected at random for our experiments. The categories are Admiral, Black Swallowtail, Machaon, Monarch, Peacock and Zebra.
- **Indian Butterfly Dataset** has been compiled as part of the present work in collaboration between the Indian Statistical Institute and the University of Otago, NZ. It is named Titli after the Hindi word for butterfly. The current version 1 has 6 classes with 60 images per class. The six butterfly classes are Papilionidae, Pieridae, Nymphalidae, Lycaenidae, Riodinidae and Hesperiidae.

C. Feature Descriptors

We have used 2 popular feature descriptors: Dense SIFT and ensemble of GIST+HoG. But it should be noted that the proposed algorithm is general and is agnostic to feature choice. A dense variant of scale invariant feature transform (SIFT) [18] is extracted and a patch size of 10×10 is chosen with overlap. Global invariant scale transform, here referred to as GIST [19], is a global feature that describes the spatial envelope of the image using directional properties. It extracts dense multi-scale overlapping patches. Histogram of oriented gradients (HoG) features [20] are extracted in a dense grid fashion in 3×3 cells which are concatenated at each grid location to generate the descriptor. The Dense SIFT features are used separately,

while the GIST and HoG features are used as a concatanated ensemble.

IV. Results and Analysis

Experimental Results. We perform clustering on each combination of descriptor, dataset and algorithm. The average percentage accuracy is presented in Table 1 for bird and butterfly species recognition. The highest results in each column are highlighted in bold. It is observed from both tables that collaborative clustering significantly outperforms its direct competitors, that is the benchmark K-means algorithm and its major variants. Also for both tasks, for the majortiy of algorithms, DenseSIFT based features yields better results than GIST+HoG. It is also noticed that collaborative clustering outperforms the original DBSCAN and also gives slight improvement overall over the recent variant DSets-DBSCAN [14]. Though the improvements are marginal, it should be noted that the proposed algorithm has a much more lightweight formulation and implementation than DSets-DBSCAN. Moreover, we also perform Wilcoxon rank test to further explore the performance of collaborative clustering vs. DSets-DBSCAN.

Statistical Analysis. Wilcoxon signed rank test is performed across both tasks (bird and butterfly recognition) between collaborative clustering and DSets-DBSCAN and presented through Table 2. The ranks (R) are allocated according to the magnitude of difference in accuracy between the two methods. If there is a tie in the absolute difference, then the rank is split between the two. For example, if there is a tie for the values for 3rd and 4th rank, then both are given 3.5 rank. The corresponding signs (S) are allocated depending on which method outperforms for that particular experimental setting. The ones for which collaborative clustering is better have sign 1. the rest have sign -1. The Wilcoxon parameter $W = \sum SR$ is calculated for the 8 pairs of values and $W = 30$. Maximum possible rank value for $n = 8$ experiments is $n(n+1)/2 = 36$. The Wilcoxon signed rank test states that the null hypothesis (collaborative clustering and DSets-DBSCAN are equally good) may be rejected (collaborative clustering better than DSets-DBSCAN) at 5% level of significance if $W \geq 30$ (2-direction) and $W \geq 30$ (1-direction). Hence, it may be concluded that the proposed collaborative clustering performs significantly better than DSets-DBSCAN on these datasets.

Fig. 2: Sample images from each of 6 classes of the species recognition datasets: (a)-(f) Ponce Birds; (g) to (l) Indian Birds; (m)-(r) Ponce Butterflies; (s)-(x) the new Indian butterflies dataset (Titli.v1).

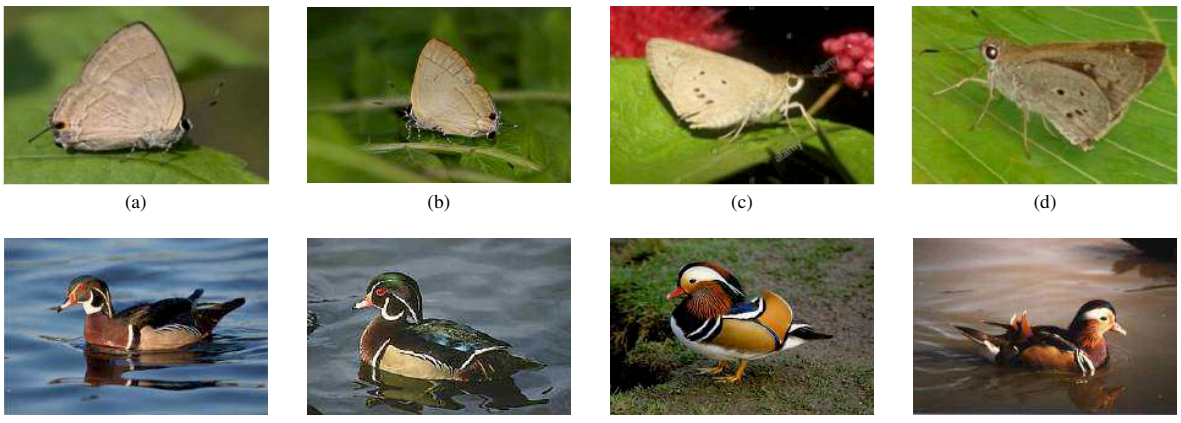
TABLE I: Clustering Accuracy %

datasets →	**Ponce Birds**		**Indian Birds**		**Ponce Butterflies**		**Indian Butterflies**	
methods ↓	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**
K-Means	72.6	73.5	70.0	72.7	67.4	69.2	65.3	68.4
K-Medians	77.7	78.4	74.3	76.6	70.8	73.9	69.9	73.5
K-Modes	77.1	78.8	74.6	76.2	71.1	73.7	70.3	73.1
K-Medoids	79.0	80.3	76.9	79.5	74.0	75.3	73.6	75.0
DBSCAN	83.6	84.1	80.5	84.8	79.4	81.8	78.5	80.8
DSet-DBSCAN	87.5	**88.6**	85.0	89.2	74.5	85.4	83.7	**85.5**
Collab. Clust.	**88.8**	88.1	**86.3**	**89.9**	**75.3**	**87.2**	**84.9**	85.2

TABLE II: Wilcoxon Signed Rank Test

datasets →	**Ponce Birds**		**Indian Birds**		**Ponce Butterflies**		**Indian Butterflies**	
methods ↓	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**	**Gist/HoG**	**SIFT**
DSet-DBSCAN	87.5	88.6	85.0	89.2	74.5	85.4	83.7	85.5
Collab. Clust.	88.8	88.1	86.3	89.9	75.3	87.2	84.9	85.2
| Difference |	1.3	0.5	1.3	0.7	0.8	1.8	1.2	0.3
Rank (R)	6.5	2	6.5	3	4	8	5	1
Sign (S)	+1	-1	+1	+1	+1	+1	+1	-1

Normalised Mutual Information (NMI). NMI is considered to be a standard procedure to investigate the performance of two closely performing clustering methods. It is given by:



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 3: Mis-clustering examples: (a)-(b) are Lycaenidae and (c)-(d) are Hesperiidae from the new Indian Butterfly dataset; these are fine-grained classes. Both DStes-DBSCAN and collaborative clustering wrongly assigned 3(c) to the Lycaenidae cluster. (e)-(f) are of Wood Duck and (g)-(h) are of Mandarin from the Ponce Duck Dataset. DSets-DBSCAN wrongly assigned 3(h) to Wood Duck cluster, but collaborative clustering correctly identified it as Mandarin.

$$NMI(Y,C) = \frac{2 \times I(Y,C)}{[H(Y) + H(C)]} \tag{8}$$

Here Y are expected/class labels and C are estimated/cluster labels. H and I are entropy and mutual information functions respectively. The entropy function $H(Y)$ is given by

$$H(Y) = -\sum_{y} P(Y = y) \times log[P(Y = y)] \tag{9}$$

The function takes the similar corresponding form for $H(C)$. The mutual information is given by

$$I(Y,C) = H(Y) - H(Y|C) \tag{10}$$

where $H(Y|C)$ is the entropy of expected labels within each cluster. Following the calculations described in [22], we find the NMI between the proposed Collaborative Clustering against the closest competitor DSet-DBSCAN and observe that

$$\frac{NMI(Y, DsetsDBSCAN)}{NMI(Y, Collab.Clust.)} < 1 \tag{11}$$

This signifies that Collaborative Clustering outperforms DSets-DBSCAN by normalised mutual information.

Qualitative Example. One of the challenges of fine-grained image categorization is utilising discriminating parts which may be obfuscated due to pose variation, bad illumination, partial obstruction by surrounding objects, etc. We have provided in Fig. 3, examples of correct and wrong performance of collaborative clustering. Fig 3. (a)-(b) are Lycaenidae and (c)-(d) are Hesperiidae from the new Indian Butterfly dataset (Titli.v1); these are fine-grained classes. Both DSets-DBSCAN and collaborative clustering wrongly assigned 3(c) to the Lycaenidae cluster. (e)-(f) are of Wood Duck and (g)-(h) are of Mandarin from the Ponce Duck Dataset. DSets-DBSCAN wrongly assigned 3(h) to Wood Duck cluster, but collaborative clustering correctly identified it as Mandarin.

V. Conclusion

Summary of Work. We present collaborative clustering as a generalization of the benchmark K-means algorithm. The contribution is to find out cluster centroids in each iteration as weighted mean of data points, where the weights are optimized using a collaborative filter. The data points are given this weighted representation with respect to the cluster centers. Thus K-means may be considered as a specialized case where the weights are unity and hence the distance from the cluster centers are Euclidean. Recent research has shown that colloborative filters are well suited in representing fine-grained image data and give good results even with limited labels/annotations. So in this work, we use the proposed collaborative clustering to categorize fine-grained species images (birds and butterflies) and compare results with K-means and its variants as well as the highly cited DBSCAN algorithm, along with its recent variant DSets-DBSCAN.

Future Work. It will be interesting to apply collaborative clustering to other similar problems as a generalised method with cluster number estimation. As expansion of this work, we also aim to investigate retrieval of poorly labeled images from large datasets. Consider the case of a specialised problem like endangered species recognition requiring labeled images from domain experts. In those cases the collaborative clustering algorithm may provide a more robust representation to retrieve similar images. We are also working on expanding our new Indian butterflies dataset to Titli.v2 by including more types and more samples per type.

References

[1] Y. Chai, "Advances in Fine-grained Visual Categorization," *University of Oxford*, 2015.

[2] E. Rodner, M. Simon, G. Brehm, S. Pietsch, J.-W. Wgele, and J. Denzler, "Fine-grained Recognition Datasets for Biodiversity Analysis," *Proc. CVPR*, 2015.

[3] T. Chakraborti and B. McCane and S. Mills and U. Pal, "Collaborative representation based fine-grained species recognition," *Proc. IVCNZ*, 2016.

[4] J. B. Schafer, D. Frankowski, J. Herlocker and S. Sen, "Collaborative Filtering Recommender Systems," *The Adaptive Web, Lecture Notes in Computer Science, Springer*, vol. 4321, pp. 291-324, 2007.

[5] L. Zhang, M. Yang and X. Feng, "Sparse Representation or Collaborative Representation: Which Helps Face Recognition?," *Proc. ICCV*, 2011.

[6] T. Chakraborti and B. McCane and S. Mills and U. Pal, "A Generalized Formulation for Collaborative Representation of Image Patches (GP-CRC)," *Proc. BMVC*, 2017.

[7] T. Chakraborti and B. McCane and S. Mills and U. Pal, "LOOP Descriptor: Local Optimal-Oriented Pattern," *IEEE Signal Processing Letters*, vol. 25, no. 5, pp. 635-639, 2018.

[8] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. in Information Thoery*, vol. 28, no. 2, pp. 129-137, 1982.

[9] A Cornujols, C Wemmert, P Ganarski and Y Bennani, "Collaborative clustering: Why, when, what and how," *Information Fusion*, vol. 39, pp. 81-95, 2018.

[10] A. Chaturvedi, P. E. Green, and D. Caroll, "K-modes Clustering," *Journal of Classification*, vol. 18, no. 1, pp. 35-55, 2001.

[11] A. K. Jain and R. C. Dubes, "Algorithms for Clustering Data," *Prentice-Hall*, 1998.

[12] H.S. Park and C.H. Jun, "A simple and fast algorithm for K-medoids clustering," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3336-3341, 2009.

[13] M. Ester, H-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Proc. Intl. Conf. on Knowledge Discovery and Data Mining*, 1996.

[14] J. Hou, H. Gao and X. Li, "DSets-DBSCAN: A Parameter-Free Clustering Algorithm," *IEEE Trans. on Image Processing*, vol. 25, no. 7, pp. 3182-3193, 2016.

[15] S. Lazebnik and C. Schmid and J. Ponce, "A Maximum Entropy Framework for Part-Based Texture and Object Recognition," *Proc. ICCV*, 2005.

[16] T. Chakraborti, B. McCane, S. Mills and U. Pal, "CoCoNet: Collaborative ConvNet for deep transfer learning of fine-grained classes," *Pattern Recognition Letters*, 2018.

[17] S. Lazebnik and C. Schmid and J. Ponce, "Semi-Local Affine Parts for Object Recognition," *Proc. BMVC*, 2004.

[18] D. G. Lowe, "Object recognition from local scale-invariant features," *Proc. ICCV*, 1999.

[19] A. Oliva and A. Torralba, "Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope," *IVCV*, vol. 42, no. 3, pp. 145-175, 2001.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proc. CVPR*, 2005.

[21] M. A. Woodbury, "Inverting modified matrices," *Memorandum report*, vol. 42, no. 106, pp. 336, 1950.

[22] N. X. Vinh, J. Epps, and J. Bailey, "Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance," *The Journal of Machine Learning Research*, vol. 11, pp. 2837-2854, 2010.